IBM IMS DataPropagator for z/OS

An Introduction

Version 3 Release 1

IBM IMS DataPropagator for z/OS

An Introduction

Version 3 Release 1

> **Note:**
> Before using this information and the product it supports, be sure to read the information in "Notices" on page 43.

Second Edition (May 2006) (Softcopy Only)

This edition applies to Version 3 Release 1 of IBM IMS DataPropagator for z/OS, 5655-E52, and to any subsequent releases until otherwise indicated in new editions or technical newsletters. This edition is available in softcopy format only. The technical changes for this edition are indicated by a vertical bar to the left of a change.

This edition replaces GC27-1211-00.

Abstract

This book provides introductory information about IMS DataPropagator for z/OS Version 3 Release 1.

Contents

Figures

Chapter 1. Preface

This book introduces IMS™ DataPropagator™ for z/OS® (IMS DPROP), a program that maintains consistency between two copies of the same data, where one copy of the data is stored in an IMS Database Manager (IMS DB) database and the other copy is stored in a DB2® or IMS database. IMS DPROP is part of IBM's data replication solution.

This book is designed to help you evaluate IMS DPROP for use in your organization.

This softcopy book is available in PDF, BookManager®, and XHTML formats. This book is available on the z/OS Software Products Collection Kit, SK3T-4270. You can get the most current versions of the PDF, BookManager, and XHTML formats by going to the IBM® Data Management Tools Web site at www.ibm.com/software/data/db2imstools-library.html and linking to the Library page.

Changes to This Book for IMS DataPropagator for z/OS Version 3 Release 1

IMS DataPropagator Version 3 contains improvements to both the product and the product library.

Product Changes

IMS DataPropagator Version 3 provides near real-time and point-in-time propagation with MQSeries-based asynchronous (MQ-ASYNC) propagation. IMS-to-IMS propagation is also now possible with MQ-ASYNC or by using a combination of the LOG-ASYNC Selector component with MQ-ASYNC functions.

Product Library Changes

The Version 3 library has been updated with information about MQ-ASYNC and IMS-to-IMS propagation. There are now three Administrators Guides, one for each primary mode of propagation. The following books are affected by these changes:
- *Administrators Guide for MQSeries Asynchronous Propagation*
- *Administrators Guide for Log Asynchronous Propagation*
- *Administrators Guide for Synchronous Propagation*
- *Messages and Codes*
- *Reference*

How This Book is Organized

The first three chapters provide a general introduction to IMS DPROP and data propagation; the next four chapters provide technical details of the product; and the last chapter provides details of the product library.

Here is a synopsis of each chapter:
- Chapter 2, "Improving Access to Information," on page 3 explains IBM's answer to managing copies of data, describes the products involved, and their interaction with each other.
- Chapter 3, "Concepts and Terminology," on page 13 explains the concepts associated with data propagation and IMS DPROP.

- Chapter 4, "Using IMS DPROP," on page 17 tells you what IMS DPROP can do for you and how it works.
- Chapter 5, "Performance," on page 29 explains performance considerations for the implementation of IMS DPROP.
- Chapter 6, "Commonly Asked Questions and Answers About IMS DPROP," on page 31 contains frequently asked questions and their answers about IMS DPROP.
- Chapter 7, "General Requirements," on page 37 describes the software and hardware requirements for IMS DPROP.
- Chapter 8, "IMS DPROP Library," on page 41 describes the contents of the books in the IMS DPROP library.

Chapter 2. Improving Access to Information

In today's business world, you need up-to-date, relevant information. This information enables you to react to market trends and to run your organization efficiently. The problem is that the information often resides in a variety of information systems that are based on different types of databases. Each system is suitable for one or more purposes, but they might not meet all the information requirements of your organization.

Operational applications handle the detailed transactions of your daily business. If you want to enter an order, an operational application records that order for you. If you want to send an invoice to a customer, an operational application checks what you have shipped to the customer and displays the appropriate invoice on the monitor.

Decision Support applications help you to manage and control your business by enabling you to analyze the current state of your business data and the trends shown in your business data through time. If you want to know what your best selling items are, a decision support application enables you to find the items with the highest sales volumes, month by month over the last year. If you want to know how efficient your handling of orders is, a decision support application enables you to find the average time interval for each stage in your shipment process.

Both kinds of applications have to be supported by reliable database management systems (DBMSs) for the storage and manipulation of large quantities of data. Two such DBMSs are Information Management System Database (IMS) and Database 2™ (DB2); both from IBM. You can use IMS for your operational applications. You can use DB2 for your operational applications and also for your decision support applications.

In some cases, the same data is accessed by two operational applications, each based on different database technology; one on IMS and the other on DB2. You need to keep a copy of the data in IMS and a copy in DB2, and you need some means of keeping the two copies identical, or synchronized. In other cases, the data accessed by a decision support application is a copy of data in an operational database. You need a means to update, or synchronize, your decision support copy on a scheduled basis; maybe nightly or weekly.

Besides your company's internal need for data that is "flexible", emerging technology is enabling the move to client/server solutions, network computing, and application designs that use distributed databases. To enable distributed data, the information must be copied (with integrity) across databases. What you need is a data replication solution.

The following topics provide additional information:
- "IBM's Data Replication Solution" on page 4
- "IMS DataPropagator" on page 4
- "Data Extract Products" on page 4
- "DB2 DataPropagator" on page 5
- "IMS DPROP Solutions" on page 6
- "Examples of Using IMS DPROP" on page 7
- "Benefits of IMS DPROP" on page 11

IBM's Data Replication Solution

IBM offers a solution to replicate (or copy) non-relational and relational data among mainframe, network, and workstation databases. This solution:

- Includes a suite of products that address the operational, informational, and administrative requirements of data replication.
- Supports propagation of data changes, as well as full refresh copying. These options combine to minimize the impact to production systems, reduce network traffic for copies, and automate the initialization and recovery process.
- Enables client/server applications and legacy data systems to work seamlessly with relational database management systems.
- Enables you to deliver key data from mission-critical operational systems (IMS data) to users across your enterprise supporting mainframe-based computing, LAN servers, and individual workstations.
- Enables you to create relational information warehouses (on the platform of your choice) from data in an IMS database.

The following products, when used together, are IBM's Data Replication solution:
- IMS DataPropagator
- Data Extract products
- DB2 DataPropagator

IMS DataPropagator

IMS DataPropagator (IMS DPROP) is the "IMS Capture" component of the Data Replication family of products. IMS DPROP is designed to maintain consistency between an IMS copy and a DB2 copy of the same data. It can also be used to maintain consistency between one or more copies of the same IMS data. This consistency is achieved by capturing changes made by an application to one type of database and applying them to the other. The details of this process are described in Chapter 4, "Using IMS DPROP," on page 17.

IMS DPROP can propagate data asynchronously from IMS to IMS or from IMS to DB2. Changes that are made to the source database are propagated when triggered by certain events or time intervals (outside the IMS application's unit of work) to the target database.

IMS DPROP can also propagate data synchronously from IMS to DB2 and from DB2 to IMS. This means changes to the source database are propagated at the same time (within the same unit of work) to the target database. Synchronous propagation in two directions between IMS and DB2 is also possible with IMS DPROP. For synchronous propagation (except for propagation via IMS batch), the IMS Transaction Manager (IMS TM) is required, in addition to the IMS Database Manager (IMS DB) and DB2.

Data Extract Products

Data extract products (such as IBM's DataRefresher™) take a full copy of the data from a data source and overwrite the data in the target database. This is known as *refresh* copying.

The extract product creates the initial DB2 copy of a database for IMS DPROP by extracting the data from the IMS database and loading that data into the DB2 database. Refer to *IMS DataPropagator for z/OS: Concepts* for details on how IMS DPROP works with DataRefresher.

DB2 DataPropagator

DB2 DataPropagator is a data replication product that captures changes made to tables in a relational database and applies changes to target tables at another relational database, either locally or remotely. Remote databases are connected using Distributed Relational Database Architecture™ (DRDA®). DB2 DataPropagator stores the changes in a data staging area (which is an openly architected interface that is managed by DB2 DPROP) prior to applying the changes to the target databases (see Figure 1 on page 6). DB2 DataPropagator can use the full power of Structured Query Language (SQL) to create a subset of the data propagated and to enhance the data.

Another way the changed data can get into the data staging area is directly from IMS DPROP. This enables the propagation of IMS data in the following two scenarios:

- DB2 DataPropagator can take the IMS data from the data staging area and propagate these changes to other relational databases. Therefore, the replication power of IMS DPROP is enhanced by extending IMS data propagation into the full relational database world, host and client/server (DB2 for Windows NT®, DB2 for OS/400®, DB2 for AIX®, HP-UX, or the Solaris Operating Environment**). Refer to Figure 1 on page 6 for an illustration of how IMS DPROP interacts with DataPropagator Relational.
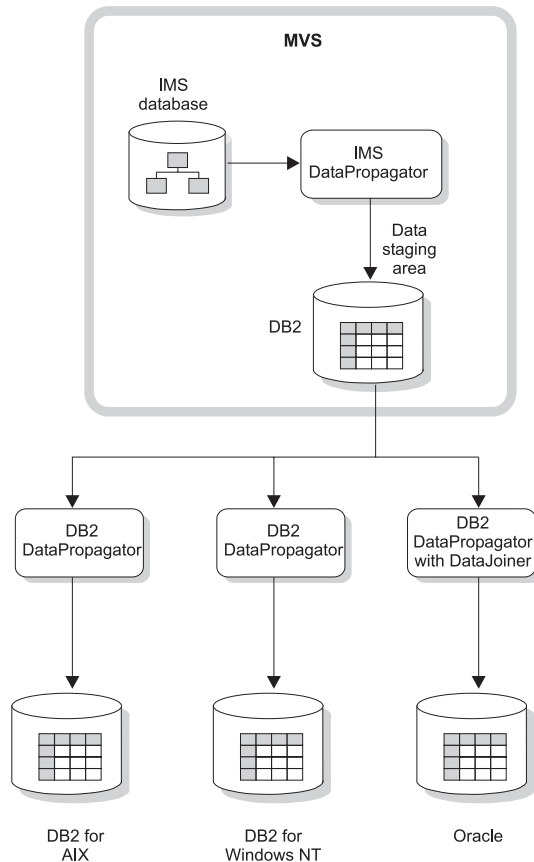
Figure 1. IMS DPROP and DB2 DataPropagator

- IMS DPROP along with DataJoiner® (another IBM product), can also take the IMS data from the staging area and propagate it to multi-vendor databases, such as, Oracle**, Sybase**, Informix®**, and Microsoft® SQL Server**.

These two scenarios further illustrate that IMS DPROP is the "IMS Capture" component of the IBM Data Replication Solution.

Storing IMS changes into the data staging area is also very useful for maintaining historical data. The changes can be used to maintain an audit trail of the changes to IMS, to do trend analysis (which would not be possible on the operational system), and to do data aggregation.

IMS DPROP Solutions

The solutions provided by IMS DPROP can be divided into 3 main areas:
    Information Access
    Coexistence
    Migration

The following sections cover these topics.

Information Access

In response to the requirement for information, you need to make operational data available to a decision support application at regular intervals. For example, you need to collect accurate and up-to-date accounting information from an IMS

database. However, the information in the operational database is not always in a suitable format and you may only need to access a subset of it.

IMS DPROP provides you with a facility to update a DB2 decision support database with data collected from your IMS operational databases. This allows you to use DB2 and the decision support tools based on it to provide your users with extensive, flexible and timely access to your business information. It requires no changes to your operational applications and has minimal impact on their performance.

In addition to providing a DB2 decision support database, suppose you want to perform trend analysis on your IMS data. Effective trend analysis requires a data history. IMS DPROP can easily and automatically create a history of changes in a DB2 table without you changing or writing a single application.

Coexistence

You have a very significant investment in your IMS operational applications. These operations are the backbone of your business; they are performing well and there is no business justification for redesigning them.

IMS DPROP allows you to maintain copies of IMS data in a DB2 environment. This is a significant benefit when you want to extend your IMS database applications with new components based on DB2 technology, but do not want to affect any existing IMS database applications. IMS DPROP protects your IMS investment while taking advantage of relational technology.

In addition, you can maintain one or more consistent copies of your IMS data in IMS databases. For example, you can retain copies of operational IMS databases at a remote site for disaster recovery purposes.

Migration

You might want to move some of your existing applications from an IMS environment to a DB2 environment. You need to implement this migration in a secure and ordered way.

IMS DPROP provides you with the facility to keep the DB2 data consistent with the original IMS data. This allows you to implement a structured, low risk migration over time. You can migrate components or programs one at a time and ensure that the migration is successful. The migrated programs will access the DB2 copy of the data through the SQL interface, while the programs not yet migrated will continue to access the IMS copy through the DL/I call interface.

Examples of Using IMS DPROP

The two scenarios that follow illustrate how IMS DPROP can help your business. These scenarios are practical examples of situations in which IMS DPROP can meet your requirements.

CelDial Corporation[1] is a company that manufactures cellular phones and sells them from sales offices across the country. The company has a manufacturing plant

---

1. The CelDial Corporation is a fictitious company used to describe the possible application of IMS DPROP.

in Washington. The phones are sold from regional sales offices based in the cities of San Jose, Chicago, Dallas, and Philadelphia. The CelDial data center is based in New York.

Orders from the regional sales offices are input over a phone link to the data center, where they are recorded in an order entry system that uses IMS. If an order cannot be met immediately from stock, it is sent to manufacturing applications, also run at the data center, to speed up the stock replenishment process. The manufacturing applications, including the stock control application, use DB2.

Let's have a look at a couple of typical situations that might arise.

Scenario 1: Decision Support Systems

At the manufacturing plant, the stock control manager feels he is getting far too many requests to speed up delivery to regional offices of items that are out of stock. He believes that a lot more orders should be met automatically from stock and would like a way of analyzing the order situation for every item sold, in order to achieve this.

The data processing manager knows that the Orders database contains records of all orders handled during the last year and she wants to get it into a form that will be of use to the stock control manager. Her plan is to create and periodically update a decision support database in DB2, the Orders Analysis database. The Orders Analysis database will be made available to all stock control personnel through an application that allows them to formulate their own queries.

Although her plan for this decision support database came about through a request to solve one problem, the data processing manager realizes that such a database could have other uses. It could allow the stock control manager and his staff to monitor and control many key indicators related to how the ordering and stock control processes are working in the company. How is this database created and how is it updated?

Initially, an extract tool (such as IBM's DataRefresher) is used to scan the IMS Orders database and extract into a file a subset of the fields from every order item segment. This file is then loaded into the DB2 Orders Analysis database using the standard DB2 Load utility. Subsequently, IMS DPROP can be used each evening to select, from the IMS logs, all the relevant changes to the Orders database and apply them to the Orders Analysis database. This is an example of log-based asynchronous propagation. Figure 2 on page 9 demonstrates how IMS DPROP supports the update of a decision support database.
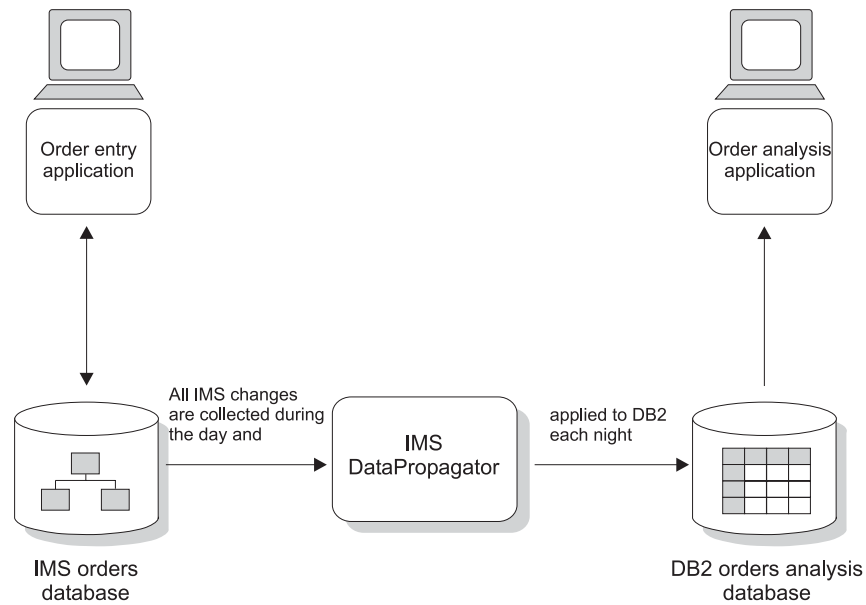
Figure 2. Decision Support Scenario (Log-Based Asynchronous Propagation)

IMS DPROP can also be used to provide near real-time updates to DB2 from changes made by IMS applications. This method involves:

- An IMS DPROP-provided exit routine to capture the IMS database changes
- The MQSeries® product to transmit the changes to the system where DB2 is running
- An IMS DPROP-provided Apply program (and other components) to update DB2

If the IMS and the DB2 subsystems are running on the same z/OS image, the time delay between the IMS update and the DB2 update can be a matter of seconds.

IMS DBMS

Application Program

IMS DPROP Capture

IMS Source DBs

MQPUT

MQSeries Queues
- **Recoverable**
- **Reliable**
- **Buffered**
- **High Performance**

Q1

Q2

Mapping Definitions

MQGET

IMS DPROP Apply Program

RUP

Other Programs

DB2/390 Target Tables

DB2/390 Staging Tables

DB2 DPROP

Nearly any relational tables
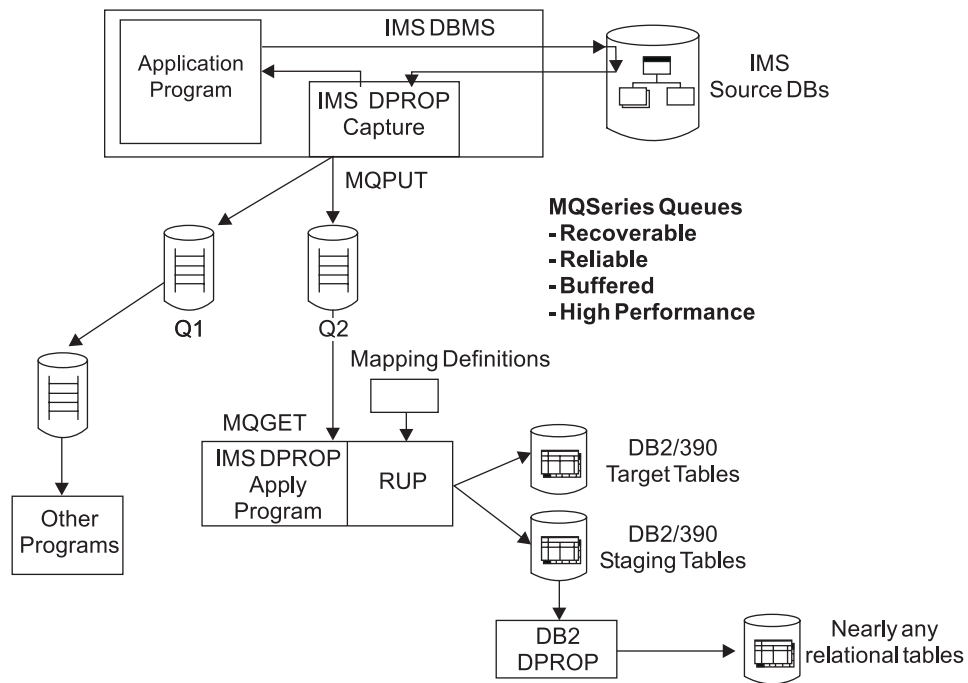
Figure 3. IMS DPROP Using MQSeries-Based Propagation

The data processing manager has a very efficient means of maintaining the Orders Analysis database; much cheaper than reloading it occasionally to keep the DB2 data current enough for her business applications. She did not have to alter any of the operational applications to achieve her goals.

Scenario 2: Extending an Application

In this scenario, the stock control manager realizes that the stock control application should take into account the backlog of orders directly. He commissions a design analysis. It concludes that detailed, up-to-date orders should be made available to the stock control application, in addition to the monthly summary of volumes of stock shipped, which is currently available to the application. It would then be possible to improve the manufacturing schedule to significantly reduce the number of times the regional sales offices are out of stock of critical items. Also, the stock control application could automatically update customers' orders when it becomes aware of new, improved items that replace existing items.

The stock control application is based on DB2, and the data processing manager decides to extend it by a new component which is also based on DB2, even though the Orders database is actually in IMS. She decides to maintain two copies of the Orders database, one (the original) in IMS and the other in DB2, and she also decides that they must be maintained so that they remain synchronized. Both must reflect the exact order situation at all times.

How can both copies be kept synchronized when updates can happen to either, and propagation must therefore be supported in two directions? It is really quite simple. IMS DPROP is designed to meet this need.

The order entry application does not need to be altered. The stock control application is extended with a new component, written as a standard DB2 and IMS application that references and sometimes updates a DB2 Orders database. No application programs reflect the fact that the orders information is duplicated. Both

copies are kept synchronized entirely by IMS DPROP. Even in the event of a
system failure, the applications are saved from design impact since recovery is
handled by standard IMS recovery mechanisms. Figure 4 demonstrates how IMS
DPROP supports the maintenance of two copies of data by simultaneously updating
both copies. This is an example of 2-way synchronous propagation.
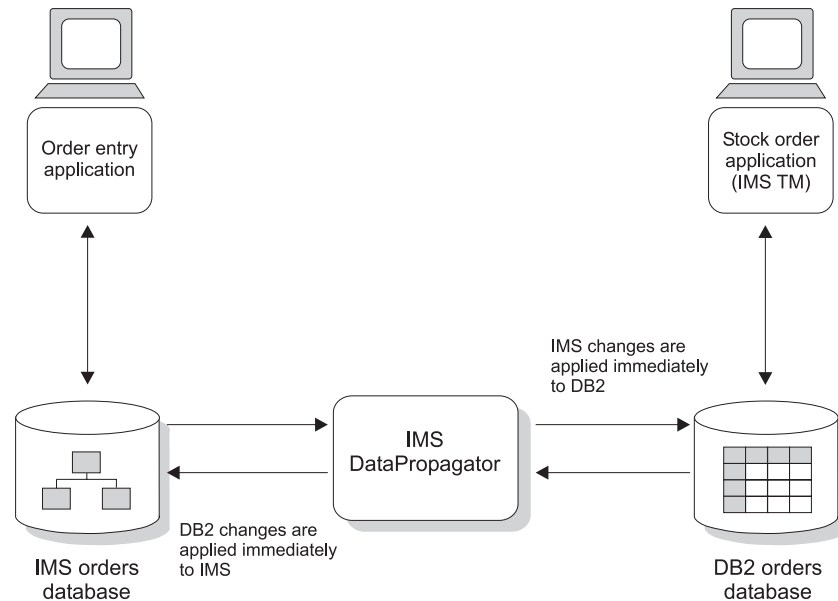
Figure 4. Extending an Application (Synchronous Propagation)

Benefits of IMS DPROP

IMS DPROP is the solution to your data consistency problems. IMS DPROP
enables you to:

- **Develop New Applications Using DB2**

  You can use DB2 technology for applications that need data currently stored in
  IMS. This means you can develop new operational and decision support
  applications using DB2 technology and data.

- **Be Flexible in Choosing a Database Manager**

  You can decide which database manager meets the various data access
  requirements of new applications or extensions to existing applications.

- **Migrate in an Orderly Fashion**

  If you plan to migrate some of your applications to DB2, you now have a
  modular, low risk way to do this over time.

- **Build New Business Solutions with Ease**

  Applications can be developed from IMS data using DB2 end user tools, such as
  Query Management Facility (QMF™). This means users with little data processing
  experience can develop applications, thereby reducing your application backlog.

- **Provide Historical Data on Updates**

  You can use the data collected for updates to analyze your business and spot
  trends.

The advantages of using IMS DPROP are:

- There is no impact on application program logic

- There is minimal impact on the performance of application programs
- You can choose update frequency and method
- It provides for recovery using standard IMS methods
- Asynchronous propagation can update local or remote copies of IMS and DB2 for z/OS
- It supports data sharing in the Sysplex environment

Chapter 3. Concepts and Terminology

You are interested in propagating data; let's take a closer look at what this means.

You have decided that you want to maintain multiple copies of some data. You know what the data is, where you want it copied to, and how frequently you want it updated. The challenge is to perform the updates with the minimum impact to your existing system while taking maximum advantage of the new possibilities available.

The following topics provide additional information:
* "Method of Copying Data"
* "Mode of Data Propagation"
* "Selecting Data for Propagation" on page 14
* "Defining the Mapping for Data Propagation" on page 14
* "Direction of Data Propagation" on page 15

Method of Copying Data

Data Propagation is a method of copying data whereby changes made to a data source are made available for updating a target database. The changes that are copied are sometimes called deltas or differentials. The propagation method of copying changes is more efficient than the refresh method because it generally involves the transport and manipulation of less data. The propagation approach is the most appropriate copy method if:

* The copy is to be updated frequently or at regular intervals.
* The percentage of records that change during one update interval is not too high.

The propagation approach is represented in Figure 5 on page 15.

Data refresh is a term used to describe the process of:

1. Extracting data from a *source* database (generally into an intermediate file)
2. Loading that data, in its entirety, into a copy, or *target* database

A refresh has to be performed to create an initial copy, regardless of the subsequent method of updating it. A refresh approach is the most appropriate copy method if:

* The copy is to be refreshed (synchronized with its source) infrequently.
* A high percentage of records are changed during one update interval.
* There is complex mapping between the copy and its source. There might be significant computational logic or a need to construct copy records by joining multiple source database records.

Mode of Data Propagation

An important consideration is how often you want to update changes. Based on this requirement, data propagation can be asynchronous or synchronous.

Asynchronous propagation means each change in the source database is stored and the update to the target is performed later. The update to the target database is done independently from the processing of the source database. When you need to batch updates and apply them to the copy only at scheduled times, asynchronous propagation can be used to update a target database with the *point-in-time* state of the source database.

13

Asynchronous propagation is used when:
- The performance impact on existing operational applications must be minimized:
- When it's important to have a static, point-in-time copy of the source data

IMS DPROP provides two methods for performing asynchronous propagation, Log-based (LOG-ASYNC) and MQSeries-based (MQ-ASYNC). By using MQ-ASYNC propagation, you can achieve near real-time propagation of the IMS updates to the target databases. For an overview of how both methods work, see Chapter 4, "Using IMS DPROP," on page 17.

Synchronous propagation means that changes are applied immediately, within the same unit of work as your application's updates. Synchronous propagation is appropriate for situations in which both copies of the data must be exactly the same at all times.

Selecting Data for Propagation

Very often, a complete copy of the source data is not required. Instead, only a subset of the source data needs to be propagated to the target.

Horizontal fragmentation describes a situation where the copy contains only selected records or rows from the source. Typically, the selection is based on some field or column values. An example of a selection value might be the copy at a specific branch must contain only those records where the branch number is 30.

Vertical fragmentation describes a situation where the copy contains only selected fields or columns from the source. For example, the copy of an orders database might need only contain order numbers, part numbers and quantities.

IMS DPROP supports both horizontal and vertical fragmentation.

There are many details you need to consider when designing a method for propagating efficiently. Clearly though, in capturing and physically transporting change records, the propagation method must be sensitive to the needs of the various copies. Propagating irrelevant changes is costly and ineffective.

Defining the Mapping for Data Propagation

When you specify a definition of your data propagation requirements, you establish a correspondence between the source data and the target data. The definition of the correspondence is called the *mapping definition*. This definition might also include information about the database type (for example, whether IMS or DB2) and information about *data enhancement*, such as:
- Computational logic to be performed on some fields
- New fields to be generated from a combination of existing fields
- Fields converted from a coded form to a textual form by means of a look-up table; for example, 'WA' converted to 'Washington'
- Summarization

The mapping definition provides the instructions for the translation (and possibly also the enhancement) of data from the source to the target database. See Figure 5 on page 15 for an illustration of how this works.
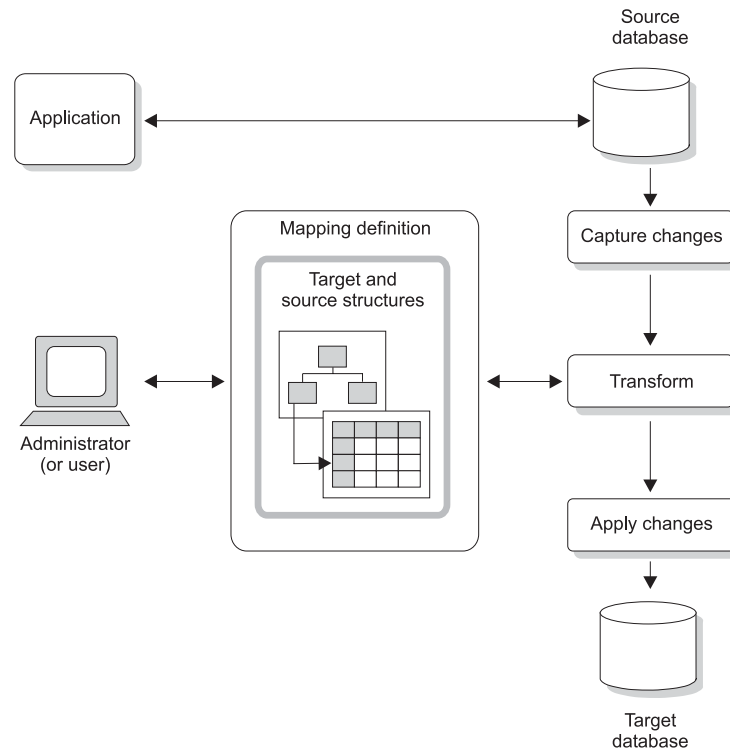
Figure 5. Data Propagation Process

Direction of Data Propagation

Depending on how you want your applications set up, you will propagate data in a certain direction.

One-way propagation is required to handle the synchronization of source/target copies of the same data, for example:

- If your application makes changes to an IMS database and you want a DB2 copy that is read-only, you will propagate one-way from IMS to DB2.
- If your application makes changes to a DB2 database and you want an IMS copy that is read-only, you will propagate one-way from DB2 to IMS.

Two-way propagation is required to handle the synchronization of two peer copies of the same data. For example, if a component of your application changes data in the IMS copy and another component changes data in the DB2 copy, you will propagate both from IMS to DB2 and from DB2 to IMS.

Chapter 4. Using IMS DPROP

IMS DPROP enables you to efficiently and securely copy the data that changed from one database to another. This chapter provides an overview of how this is done.

For a detailed description of how IMS DPROP operates, see the *IMS DataPropagator for z/OS: Concepts*.

The following topics provide additional information:
- "Overview"
- "Asynchronous Propagation (IMS-to-DB2 or IMS-to-IMS)" on page 18
- "Synchronous IMS-to-DB2 Propagation" on page 22
- "Synchronous DB2-to-IMS Propagation" on page 23
- "Synchronous Two-Way Propagation" on page 24
- "Mapping Definition" on page 24

Overview

The main IMS DPROP functions are as follows:

- **Asynchronous Propagation**

  You can use IMS DPROP to provide asynchronous updates to an IMS or DB2 database, reflecting changes that have taken place in selected IMS databases. The source and target databases can be on the same local system, or on different systems where one system is remote and the other is local. You can set the update frequency according to your business' requirements.

  There are two possible asynchronous solutions that come with IMS DPROP; log-based asynchronous (LOG-ASYNC) propagation and MQSeries-based asynchronous (MQ-ASYNC) propagation. "Asynchronous Propagation (IMS-to-DB2 or IMS-to-IMS)" on page 18 contains more details about these two methods.

- **Synchronous Propagation**

  In synchronous mode, IMS DPROP supports IMS-to-DB2, DB2-to-IMS, and two-way propagation between IMS and DB2. You can activate or deactivate propagation in a granular fashion, that is, without affecting the entire propagation process.

  Note: The DB2 and IMS databases must be on the same local system (or Sysplex), and, for synchronous propagation, IMS TM is required.

- **Mapping Definition Creation**

  You can use IMS DPROP to propagate changes between selected IMS databases, segments, fields, and selected DB2 databases, tables, and columns. In order to do so, you need to specify the relationship between the IMS data and the DB2 data. You specify the relationships in a *mapping definition*.

  You can create the mapping definitions with functions shipped with IMS DPROP.

- **Consistency Checking**

  You can use IMS DPROP to check that the data in IMS and DB2 is synchronized.

Table 1 on page 18 describes the IMS and DB2 environments that IMS DataPropagator Version 3 supports.

Table 1. IMS and DB2 Functions that IMS DataPropagator Version 3 Supports

Function	Asynchronous Mode	Synchronous Mode
IMS DB support		
Full Function DL/I	Yes	Yes
Fast Path DEDBs	Yes	Yes
IMS data sharing support		
Non-Sysplex	Yes	Yes
Sysplex	Yes	Yes
IMS DB updates from:		
IMS TM	Yes	Yes
IMS Batch (BMP)	Yes	Yes
IMS Batch	Yes (LOG-ASYNC only)	Yes
CICS®	Yes (via DBCTL)	No
Propagation direction:		
IMS to DB2	Yes	Yes
DB2 to IMS	No	Yes
IMS to IMS	Yes	No
Bidirectional	No	Yes
Transactional consistency	Yes	Yes

Asynchronous Propagation (IMS-to-DB2 or IMS-to-IMS)

Before beginning asynchronous propagation, you must extract data from the IMS source and load it into the IMS or DB2 target. For IMS-to-IMS propagation, you can take an Image Copy of the IMS source and use it to load the IMS target. For IMS-to-DB2 propagation, use an extract tool (such as DataRefresher) to extract the IMS source data for loading to the DB2 target.

The Database Administrator (DBA) first identifies the IMS database for which changes are to be captured and propagated. Typically with IMS-to-DB2 propagation, the DBA will further identify which segments are to be captured and propagated. For IMS-to-IMS propagation, the entire IMS database is usually propagated. The DBA then sets the EXIT= parameter in the database definition (DBD) that causes changes to the identified IMS database (or segments for IMS-to-DB2 propagation) to be written as MQSeries messages, to be written to the IMS log, or both, whenever a database segment is changed.

During propagation, changes that are made by applications to the source IMS database that are captured as MQSeries messages can be applied to the IMS target database or DB2 target tables for near real-time propagation as soon as they are available. These changes can also be saved and processed at a later time to effect point-in-time propagation. During propagation, changes that are made by applications to the source IMS database that are captured to the IMS log are first accumulated. After these changes are available (through an IMS archived or batch log), they can be applied to the IMS or DB2 target. IMS application programs can run as MPPs, BMPs, IFPs, or batch programs. For IMS databases, changes that are captured to IMS log applications can also run as Customer Information Control System (CICS) programs (in a DBCTL environment).

"MQ-ASYNC Propagation (IMS-to-IMS)" on page 19 and "MQ-ASYNC Propagation (IMS-to-DB2)" on page 20 describe the two asynchronous propagation methods that are available with IMS DPROP and show how to use MQ-ASYNC for IMS-to-IMS and IMS-to-DB2 propagation.

MQ-ASYNC Propagation (IMS-to-IMS)

Figure 6 illustrates the IMS DPROP method of propagation that involves the IBM MQSeries product where the IMS source and IMS target databases are on different z/OS images. By using the MQ-ASYNC method of propagation, you can achieve near real-time propagation of the IMS source updates to the IMS target databases.

Asynchronous, Near Real-Time, MQSeries Based, IMS-to-IMS Propagation

Figure 6. MQSeries-Based Asynchronous Propagation (IMS-to-IMS)

The following list describes Figure 6:

1. When an IMS application changes an IMS segment of a database that is to be propagated, IMS DPROP captures the update by using a Capture exit routine. This *capture* happens during the unit of work of the IMS application. Control is passed to the Capture exit routine after the update to the segment is made.

2. The Capture exit routine builds an MQSeries message that contains the information about the changed segment and places this message on an MQSeries message queue. The Capture exit routine can place the messages on multiple queues, as shown in the figure.

3. If all the processing is performed without any problems (the update to the segment, the building of the message, and the placing of the message on the MQSeries queue), the update to the source segment is committed and the IMS applications unit-of-work is complete.

4. An IMS DPROP IMS Apply program retrieves the message from the transmitted MQSeries queue and updates the target IMS database. As shown in Figure 6, there can be multiple instances of the IMS Apply program, each one processes

work transmitted and has only one MQSeries queue as a result of IMS updates made and captured by the IMS DPROP Capture program. The Capture exit routine and the IMS Apply program can be located on:

- The same MVS™ image (or Sysplex)
- MVS images that share DASD resources
- Sites remote from one another

MQ-ASYNC Propagation (IMS-to-DB2)

Figure 7 illustrates the IMS DPROP method of propagation that involves the IBM MQSeries product where IMS and DB2 are on different z/OS images. By using the MQ-ASYNC method of propagation, you can achieve near real-time propagation of the IMS updates to DB2.
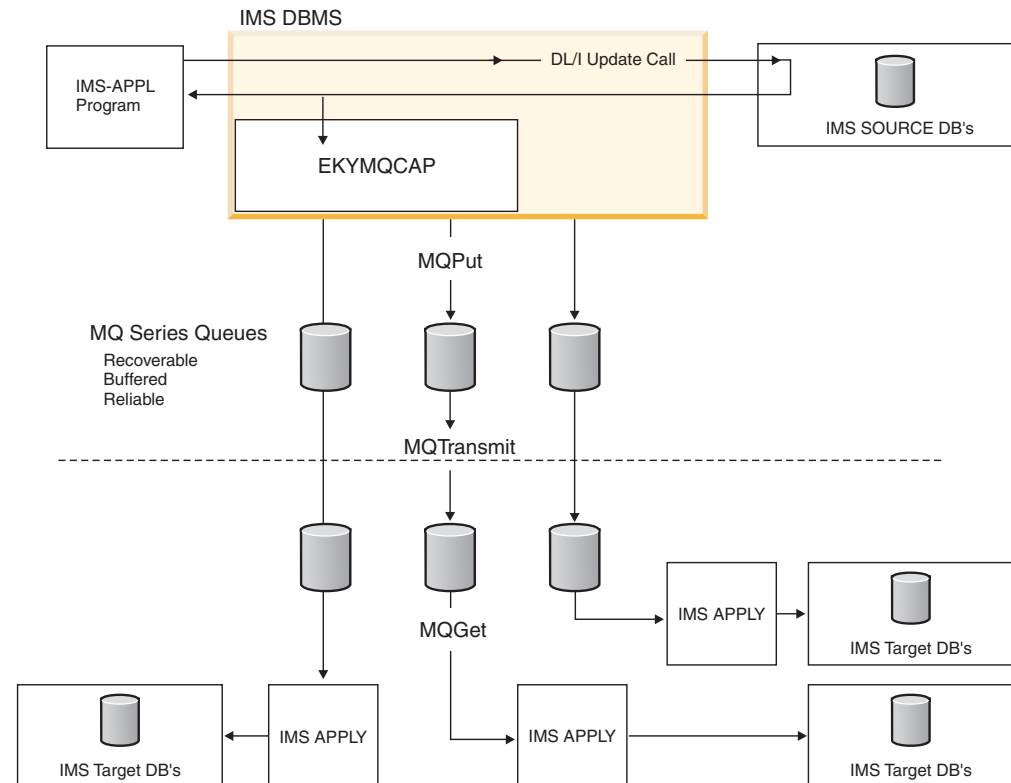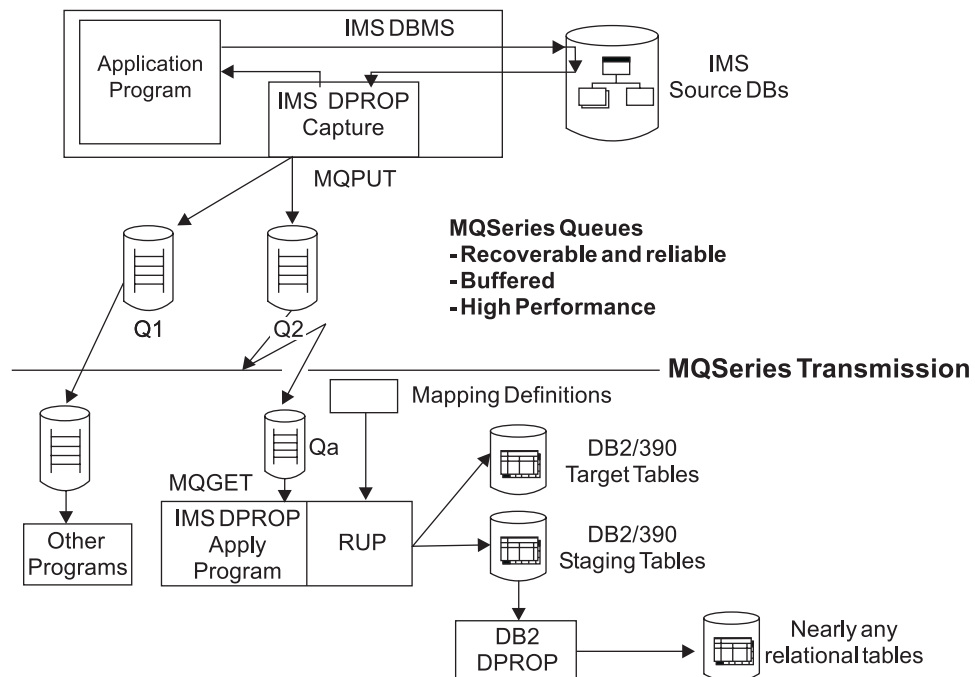

Figure 7. MQSeries-Based Asynchronous Propagation (IMS-to-DB2)

The following list describes Figure 7:

1. When an IMS application changes an IMS segment type that is to be propagated, IMS DPROP captures the update by using a Capture exit routine. This *capture* happens during the unit of work of the IMS application. Control is passed to the Capture exit routine after the update to the segment is made.

2. The Capture exit routine builds an MQSeries message that contains the information about the changed segment and places this message on an MQSeries message queue (labeled Q2 in Figure 7). The Capture exit routine can place the message on multiple queues, also shown in Figure 7.

3. If all of the processing steps complete with no problems (the update to the segment, the building of the message, and the placing of the message on the MQSeries queue), the update to the segment is committed. At this point in time, the IMS application's unit-of-work is complete.

4. The IMS DPROP Apply program retrieves the message from queue Q2 and passes it to the IMS DPROP Relational Update Program (RUP®).

5. The RUP queries the mapping definitions that describe how the IMS change data "maps" to the target DB2 tables.

6. The RUP uses the information from the mapping definitions, to update the target DB2 tables.

7. If the target DB2 tables are staging tables, the DB2 DataPropagator product can propagate the change data to almost any other relational database.

The Capture exit routine and the Apply program can be located on:
- The same MVS image (or Sysplex)
- MVS images that share DASD resources
- Sites that are remote from one another

Requirement: The Capture exit routine must be located on the same subsystem that IMS is located on and the Apply program must be located on the same subsystem that DB2 is located on. However, the Capture exit routine and the Apply program can be located on separate, or remote sites from each other.

LOG-ASYNC Propagation

Figure 8 on page 22 shows the processing that occurs during log-based asynchronous (LOG-ASYNC) data propagation. Changes to specified segment types (defined in the IMS DBD) are written by the IMS Asynchronous Changed Data Capture exit to the IMS log.

The Selector component (provided by IMS DPROP) can be run at a later point in time to gather the changed data from the IMS log data sets. The Selector uses the IMS Database Recovery Control (DBRC) interface to locate logs containing records that are relevant to propagation (propagation records). The propagation records are stored by IMS DPROP in Propagation Record Data Sets (PRDSs).

The Receiver component (provided by IMS DPROP) reads the changed data from PRDSs. It then issues a static SQL call, which updates the DB2 table.

With IMS DPROP, you have the option of using the IMS Archive utility to improve propagation performance. This utility can select propagation log records at archive time. It stores this data on Changed Data Capture Data Sets (CDCDSs). The Selector can take data from CDCDSs, instead of IMS log data sets, when it is producing PRDSs. This increases the efficiency of the Selector and of propagation.

The Selector and Receiver programs can be located on:
- The same MVS image (or Sysplex)
- MVS images sharing DASD resources
- Sites remote from one another

Note: The Selector must be located on the same subsystem that DBRC is located on and the Receiver must be located on the same subsystem that DB2 is located on. However, the Selector and Receiver can be located on separate, or remote sites.

If a Selector or Receiver is at a remote site, you must have a transport mechanism to transport the PRDSs. You provide the transport mechanism to suit your own requirements.

MVS					
IMS	IMS MPP	IMS BMP	IMS IFP	IMS Batch	CICS DBCTL

DL/1 Update

IMS application program

Capture changes

IMS Asynchronous Changed Data Capture exit

IMS database

Save changes

IMS log

IMS DataPropagator Selector

PRDS

DB2 on MVS	

Propagate data

DB2 tables

IMS DataPropagator Receiver

Static SQL statements
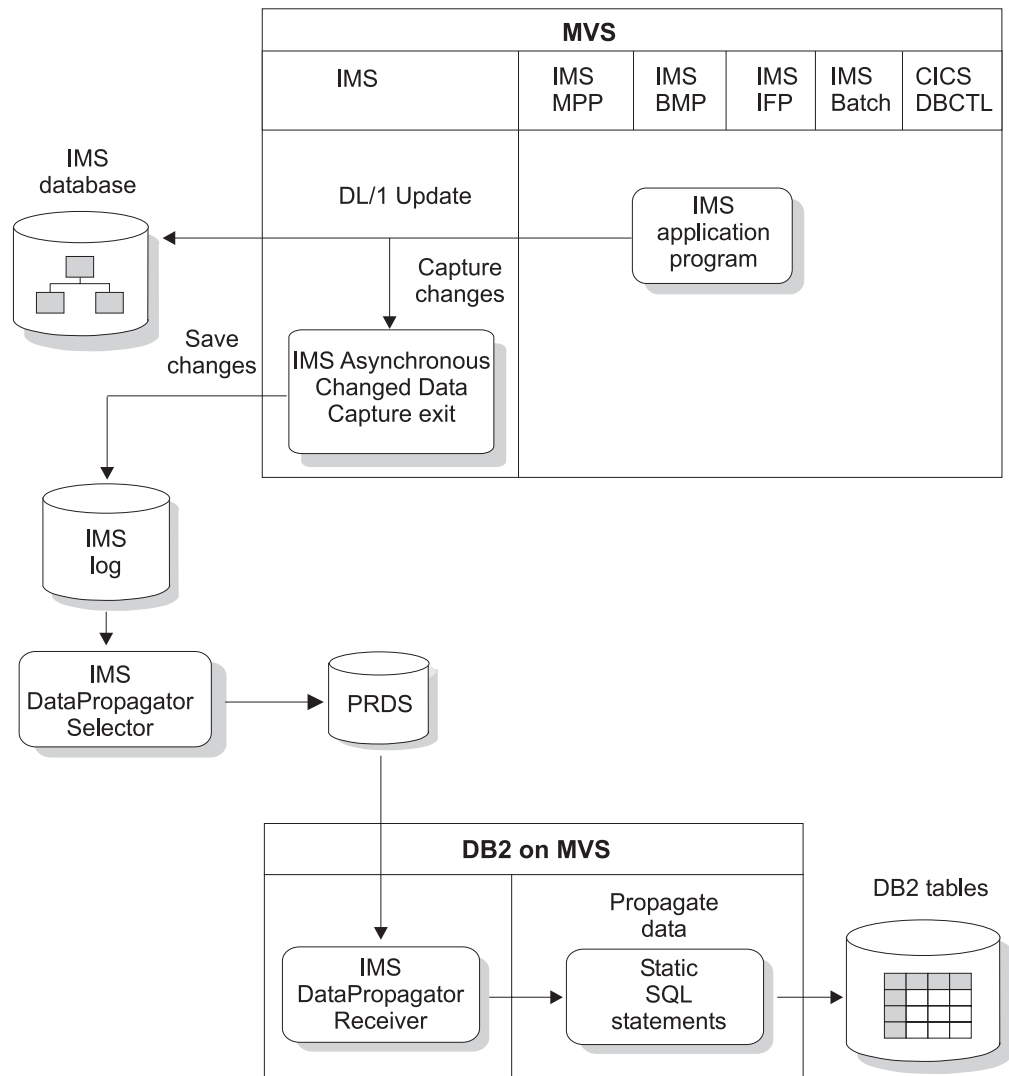
Figure 8. Log-Based Asynchronous Propagation

Synchronous IMS-to-DB2 Propagation

First, before synchronous IMS-to-DB2 propagation can begin, you must extract the data from IMS and load it into DB2. You can use DataRefresher to extract the data.

Second, the DBA defines the IMS database segments for which changes are to be captured and propagated.

During propagation, changes that are made by applications to the IMS database are copied to DB2 tables. IMS application programs can run as MPPs, BMPs, IFPs, or batch programs.

Figure 9 on page 23 shows the processing that occurs during synchronous data propagation.
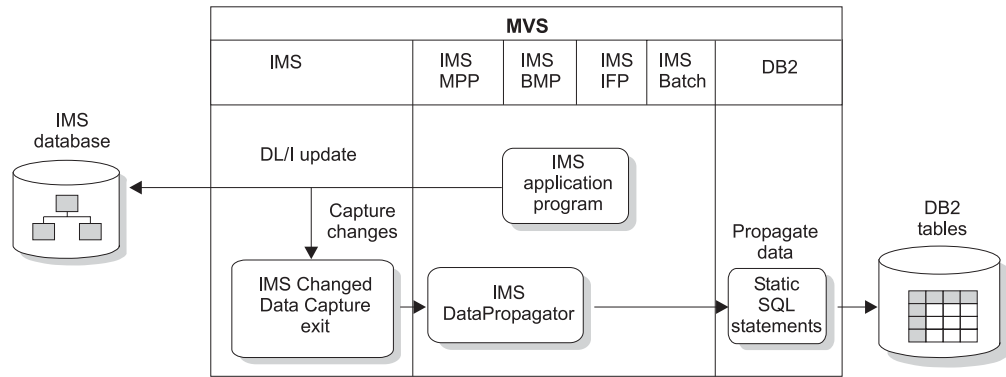
Figure 9. Synchronous IMS-to-DB2 Propagation

When an IMS segment subject to propagation is successfully changed, the IMS Changed Data Capture exit calls IMS DPROP with the changed data. After a successful change and before control is returned to the application program (that is, within the same unit of work), the data is collected and passed to IMS DPROP, which then issues a static SQL call.

Synchronous DB2-to-IMS Propagation

IMS DPROP supports DB2-to-IMS propagation only for updating SQL statements that are issued in an IMS environment through DB2's IMS attachment facility.

First, before synchronous DB2-to-IMS propagation can begin, you must extract data from DB2 and load it into IMS. You can do this using the DL/I Load utilities provided with IMS DPROP.

Second, the DBA defines the DB2 database tables for which changes are to be captured and propagated.

During propagation, changes that are made by application programs to DB2 tables are propagated to IMS databases.DB2 application programs can run as MPPs, BMPs, IFPs, or batch programs.

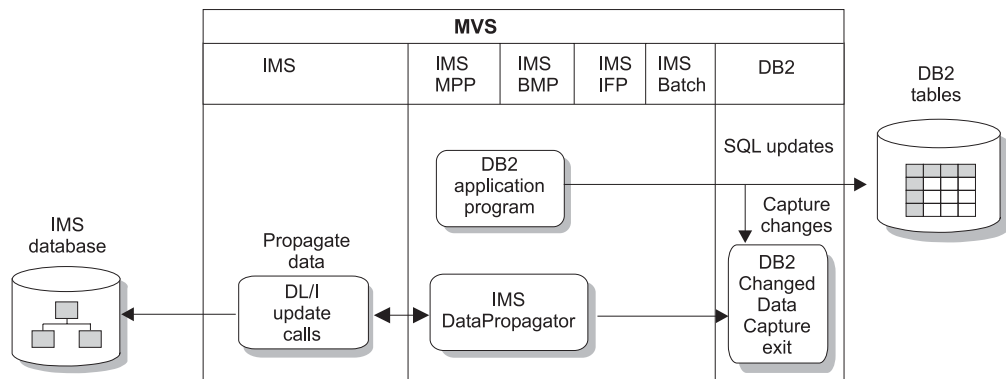Figure 10 shows the processing that occurs during synchronous DB2-to-IMS data propagation.

Figure 10. Synchronous DB2-to-IMS Propagation

This is very similar to what happens during synchronous IMS-to-DB2 propagation. When a DB2 table (defined with the DATA CAPTURE CHANGES attribute) is successfully changed, the DB2 Changed Data Capture exit notifies IMS that the DB2 table changes have been captured on the DB2 log. This signal to IMS causes IMS DPROP to be started. IMS DPROP obtains the captured DB2 data from DB2 and applies the changes to the IMS databases.

Synchronous Two-Way Propagation

Two-way propagation enables you to combine both types of synchronous propagation. You can have synchronous propagation from IMS to DB2 **and** synchronous propagation from DB2 to IMS. This makes the two synchronized database copies available for updating at the same time.

Two-way propagation enables the virtual coexistence of:
- Applications that have DL/I update access
- Applications that have SQL update access

Taking this thought one step further, two-way propagation enables
- DB2 applications to access data propagated from IMS (in the DB2 environment)
- IMS applications to access data propagated from DB2 (in the IMS environment)

One consideration about two-way propagation is that there is no primary copy of the data. This has a disadvantage. If the two copies of the data become inconsistent (for example, because of operational errors), it is difficult to resynchronize them. You have to decide if your application environment can handle situations with no primary copy of the data. If it can, you must develop procedures for resolving inconsistencies.

Mapping Definition

When data is propagated between IMS and DB2, data from keys and fields in an IMS segment must be associated with, or mapped to, rows and columns in a DB2 table. A *mapping case* defines how data is to be mapped between an IMS segment and a DB2 table. IMS DPROP provides three different types of mapping cases, called mapping case 1, 2, and 3.

IMS Database   **DB2 Tables**
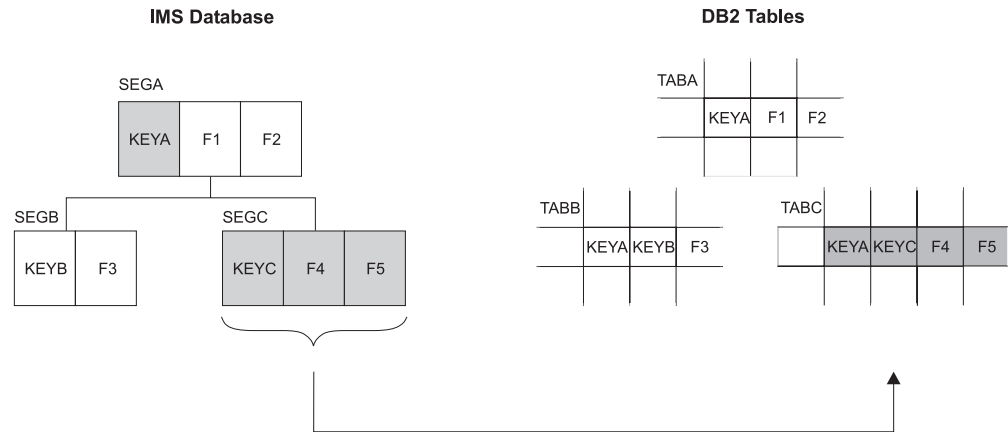
Notes:

1. **Mapping case 1** maps one single segment type occurrence with the keys of the parent and all ancestors up to the root.

2. In this figure, an occurrence of IMS segment SEGC is mapped to a row of the DB2 table TABC/D. In the table, KEYC, F4, and F5 are the key and nonkey fields from SEGC; KEYA is the key of parent SEGA.

3. If the propagation was from DB2 to IMS, the arrow in this figure would be reversed.

Figure 11. IMS DPROP Mapping Case 1

IMS Database   **DB2 Tables**

SEGA

| KEYA | F1 | F2 |

SEGB   SEGC

| KEYB | F3 | | KEYC | F4 | F5 |

SEGD

| F6 | F7 |

TABA

| KEYA | F1 | F2 |

TABB

| KEYA | KEYB | F3 |

TABC/D

| KEYA | KEYC | F4 | F5 | F6 | F7 |
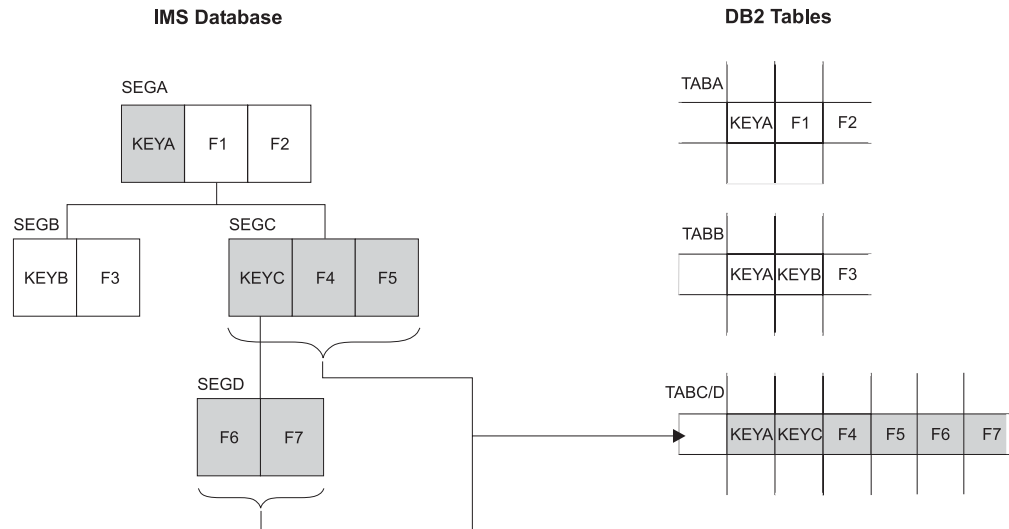
Notes:

1. **Mapping case 2** maps one single segment type occurrence with the keys of the parent and all ancestors up to the root **plus** data from one or more immediately subordinate segment types (with a maximum of one occurrence of each segment type per parent).

2. In this figure, an occurrence of IMS segment SEGC and SEGD are mapped to a row of the DB2 table TABC/D. In the table, KEYA is the key of parent SEGA. KEYC, F4, and F5 are the key and nonkey fields from SEGC. F6 and F7 are nonkey fields from the immediately subordinate segment SEGD, which occurs only once.

3. If the propagation was from DB2 to IMS, the arrow in this figure would be reversed.

Figure 12. IMS DPROP Mapping Case 2

IMS Database   **DB2 Tables**
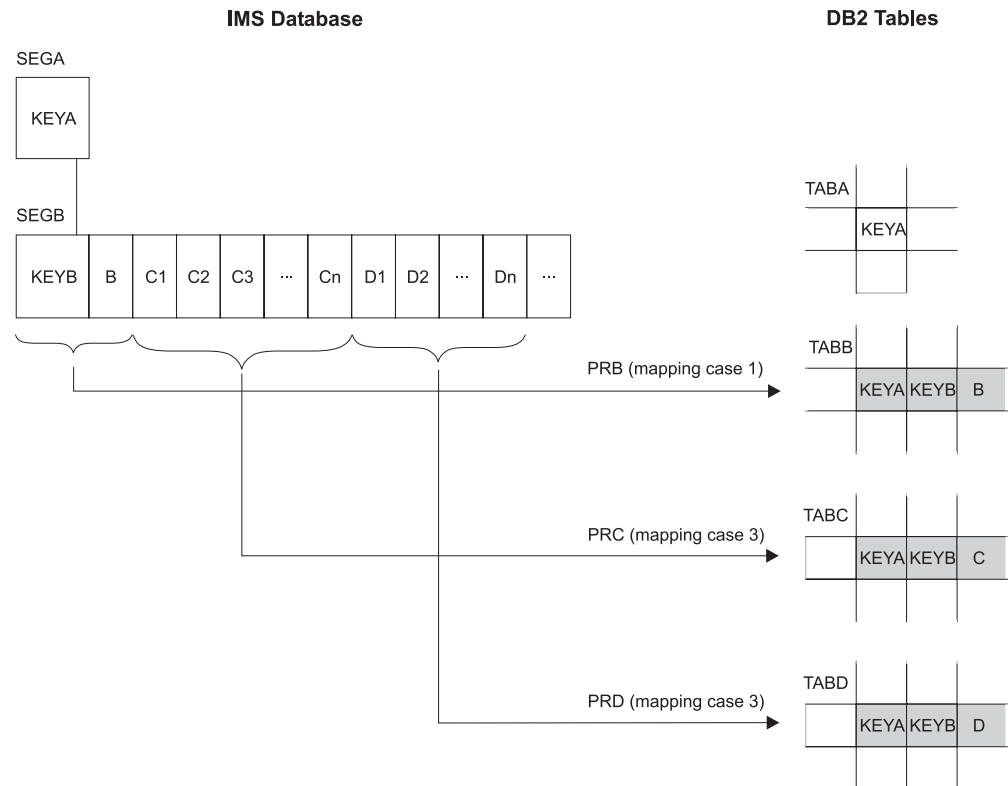
Notes:

1. **Mapping case 3** maps embedded structures. Each occurrence of an embedded structure is propagated together with the keys of the physical parent and ancestors up to the root.

2. In this figure, IMS segment SEGB contains 2 embedded structures; C and D. C and D occur multiple times within SEGB.

   The figure shows 3 different propagation requests:
   - PRC is for mapping case 3. Each occurrence of embedded structure C is propagated to one row of TABC, together with the key of segment SEGB and the keys of its physical parent and ancestors, up to the root.
   - PRD is also for mapping case 3. Propagation is similar to PRC except that embedded structure D is propagated to a different DB2 table, TABD.
   - PRB is for mapping case 1. The portion of IMS SEGB that did not belong to an embedded structure (together with key fields from each segment in the hierarchical path) is propagated to one row of TABB.

3. If the propagation was from DB2 to IMS, the arrows in this figure would be reversed.

Figure 13. IMS DPROP Mapping Case 3

IMS DPROP provides two *mapping options* that can be used with the three mapping cases. The first is the WHERE clause option. With it, you can specify under which conditions a changed IMS segment is to be copied to DB2 for IMS-to-DB2 propagation. The second is the PATH data option. With the PATH option, you can map nonkey fields of the physical parent and ancestors of an IMS segment to a DB2 table.

If your mapping requirements are not met by the IMS DPROP generalized mapping cases, you can write *exit routines* to help meet your specific requirements. There are three types of exit routines you might need to write:

- Segment exit routine, which enables you to define a segment differently from the source segment
- Field exit routine, which enables you to perform field conversions other than those supported by IMS DPROP
- Propagation exit routine, which enables you to do your own mapping and propagation

If you write your own Propagation exit routine, you can still take advantage of many IMS DPROP services, such as: tracing, standardized error processing, and audit trail generation.

You can write Segment, Field, and Propagation exit routines in high level languages, such as COBOL, PL/I, and C. The high level language support that comes with IMS DPROP requires Language Environment/370 (LE/370), as well as the compilers that support this environment.

Chapter 5. Performance

IMS DPROP has a minimal affect on the performance of your existing applications.

The affect of data propagation on performance is installation-dependent. In many cases, the additional I/Os and CPU time required for data propagation will not affect IMS or DB2 response time. Other cases do incur a performance impact.

The performance impact, in general, depends mainly on:
- The number of changes a batch program or transaction performs per execution.
- The number of those changes that are to be propagated.

Another factor is whether you are performing asynchronous or synchronous propagation. Asynchronous updates are performed **outside** the unit of work in the application, therefore affecting your performance less than synchronous updates, which are done **within** the unit of work of the application.

You can perform some calculations to help you to determine the effect of propagation on the overall system. For example:
- Estimate the percentage of the total IMS changes to data that will be propagated at any one time. For a typical IMS installation (with a large number of databases and segments), you will probably find that less than 10 per cent of the total changes to data are propagated at any given point in time.
- Estimate average ratio of update to read operations that are performed on the database. Studies show that the average ratio of update to read operations is less than 1:5.

These results would indicate that less than 2 per cent of database activity may actually cause propagation. Activities that do not affect databases, such as managing communications, do not cause propagation.

IMS DPROP ensures that the performance cost of this small percentage of activity is increased by the smallest amount. This ensures the least possible effect on the source system.

The following topics provide additional information:
- "Asynchronous Propagation"
- "Synchronous Propagation" on page 30

Asynchronous Propagation

With asynchronous data propagation, DB2 updates are performed outside the unit of work of the IMS application changes. You have more control over the potential performance impact, since you can have DB2 updates performed at times other than peak usage. You can schedule these DB2 updates as a batch job.

MQ-ASYNC Propagation Performance

With MQ-ASYNC propagation, the propagation targets can reflect changes to the source IMS databases much faster than the LOG-ASYNC method of propagation can. If conditions are optimal, target copies can reflect the IMS source updates in near real-time.

29

Because the Capture component, the MQSeries product, and the Apply component run asynchronously to each other, each one can run at its own maximum speed, without being slowed down by the other two.

MQ-ASYNC exploits high performance messaging services of the MQSeries product. To tune the performance of MQSeries, refer to MQSeries documentation. Some of factors that you should consider:

- MQ Logging
- MQ Queue Buffers
- Transmission specific parameters in MQSeries, like the number of messages per transmission unit of work and VTAM® and TCP/IP parameters (such as RU size and compression)
- Dispatching priority of the MQSeries Queue Manager

LOG-ASYNC Propagation Performance

The IMS Asynchronous Changed Data Capture exit captures changed segments to the log. The main performance impact on your applications is in writing the additional log records. These additional log records are only for the subset of data that you are interested in propagating. If the performance of your application is constrained by the amount of IMS logging, this performance impact is worth considering. In most cases, the impact is negligible.

One way you can improve performance is to use the IMS Archive utility to store the changed data that is to be propagated on changed data capture data sets (CDCDSs). The Selector component of IMS DPROP can use the CDCDSs for initializing propagation requests instead of the IMS system log data sets (SLDSs). The CDCDSs are usually much smaller than the SLDSs, which results in significant time savings when the Selector processes the CDCDSs, as compared to the SLDSs.

Synchronous Propagation

With synchronous IMS-to-DB2 data propagation, the performance impact on the IMS applications results mainly from the SQL calls issued by IMS DPROP to propagate the changed data. DB2 updates are done within the same unit of work as the IMS application; therefore, DB2 performance affects the IMS application performance. This performance impact, however, is equivalent to adding a well-tuned application into the processing loop of the original IMS application.

IMS DPROP minimizes the impact by the following techniques:

- Using static rather than dynamic SQL statements to propagate IMS database changes to the target DB2 tables.
- Exploiting the high-performance services of the MVS Virtual Lookaside Facility (VLF). (Use of VLF, for example, greatly reduces the number of SQL calls that are required to access information in theIMS DPROP directory.)
- Preserving the SQL update modules and control information in memory. This eliminates the need to use the VLF or DB2 database for reference.

Chapter 6. Commonly Asked Questions and Answers About IMS DPROP

The following questions and answers were compiled by the IMS DPROP development staff and service staff. The topics discussed are those that are asked about most often.

Why do I need an extract product like DataRefresher if I want to perform change capture for IMS?
> There are two reasons why DataRefresher complements IMS DPROP.
>
> 1. IMS DPROP captures changes and applies those changes, but does not handle the initial full refresh. This is a difference between IMS DPROP and the DPROPR apply process. You can use DataRefresher to perform the initial full refresh or a subsequent full refresh during a recovery situation.
>
> 2. You can use DataRefresher to create mappings for IMS DPROP. DataRefresher's extract mappings and IMS DPROP's mappings are consistent with each other. Any field or segment level exits are compatible between DataRefresher and IMS DPROP. Although IMS DPROP can map specifications without DataRefresher, this is a fairly labor intensive process. This is especially true when determining the offset and data types for each field to be propagated. This can be done with a batch MVS process or with the IMS DPROP ISPF panels.

Can IMS DPROP be used in a data-sharing environment?
> Yes. When using IMS DPROP in LOG-ASYNC mode, the Selector component gets all the information about the input IMS logs by interfacing with DBRC. This includes both online and batch logs in any data-sharing configuration.
>
> In synchronous mode, IMS DPROP can support a Sysplex data-sharing configuration, where both IMS and DB2 are sharing data.IMS DPROP also supports a non-Sysplex IMS data-sharing environment that is contained within a single MVS image.

What kind of data transformations does IMS DPROP support?
> You can do almost any kind of data transformation. While simple data transforms (such as decimal to small integer) are handled directly by IMS DPROP, other transformations (such as Julian date to DB2 data format) are handled through field exits. IMS DPROP supports exit routines written in COBOL, PL/I, C, and Assembler. Many sample exit routines, including the Julian date example, are provided with IMS DPROP (in all the supported languages), and can be readily altered for desired transformations.
>
> Exit routines can also be written in COBOL II, as long as the conditions listed in the next question are met.

Can exit routines be written in COBOL II, instead of"standard" COBOL and LE/370?
> Yes. IMS DPROP supports asynchronous exit routines that are written in COBOL II, as long as they include the proper run-time libraries for their COBOL version. Synchronous mode exit routines, however, require standard COBOL with LE/370.

Can IMS DPROP handle DL/I data that is compressed?
> Yes. When propagating in LOG-ASYNC mode, IMS DPROP reads log records (9904s), which are compressed by IMS. IMS DPROP contains code

31

that understands this compression, therefore, the compression does not cause problems for IMS DPROP. When propagating in MQ-ASYNC and synchronous modes, IMS DPROP is passed data that has already been expanded by IMS and, therefore, does not cause any problems. IMS DPROP is also not affected by vendor compression routines.

What environments are supported for asynchronous propagation?
IMS batch, IMS TM/DB, and CICS/DBCTL. For version information, see Chapter 7, "General Requirements," on page 37.

IMS DPROP does not support CICS with local DLI (an older way of interfacing CICS with IMS) for either asynchronous or synchronous propagation.

What are the performance characteristics of asynchronous propagation? What effect does it have on my IMS system?

MQ-ASYNC Propagation
MQ-ASYNC propagation is much faster than LOG-ASYNC propagation and can be used to achieve near real-time propagation. See "MQ-ASYNC Propagation Performance" on page 29 for more details.

LOG-ASYNC Propagation
For LOG-ASYNC propagation, the Selector adds some processing time in order to collect, merge, and sort the log records. The Receiver adds processing time in proportion to the performance cost of each static SQL call, plus a certain amount of overhead for log record translation.

LOG-ASYNC propagation is **not** a near, real-time propagation. First, the data is not available from IMS until the IMS online log has been archived. The time will vary from system to system, but generally, it will be no less than a 10-minute interval between archives plus the minutes spent performing the archive. Add to that many more minutes to run the Selector plus Receiver to process the latest log records that are available after the archive. If you are propagating data to systems other than DB2 for MVS (via DB2 DPROP), there is additional time to process the data, to deliver it, and apply it to the final platform destination.

For most customers, LOG-ASYNC propagation is most easily and efficiently set up as a once per day operation or as an automatic operation at the termination of each archive job.

IMS DPROP does not require you to change your applications or procedures when running LOG-ASYNC propagation. Optionally, you can force archives to occur more often than normal in order to speed the process of propagation.

The Selector does not affect the performance of IMS resources and the time that IMS uses to write an additional log record is insignificant compared to the IMS transaction speed.

How does synchronous propagation work?
Synchronous propagation can work in two ways:

- **Synchronous forward - IMS to DB2**

  In IMS, the Change Data Capture exit enables a user to introduce code **within** the IMS transaction when a particular IMS segment gets updated. IMS DPROP synchronous propagation processing provides the code to

be run under this exit capability. This code translates the information passed to the exit about the IMS update into an SQL insert, update, or delete call (one or more calls as needed) and performs the call.

This is a two-phase commit process, such that when the IMS update is committed, the DB2 update is also committed.

- **Synchronous reverse - DB2 to IMS**

    In DB2 Version 3, a Change Capture exit was introduced. Supporting IMS code came out with IMS Version 4. The DB2 Change Capture exit actually runs in the IMS address space. When an update of a particular DB2 table occurs, the change is logged on the DB2 log (full-row logging is required). This change is signalled back to the IMS address space. IMS then calls the DB2 Change Capture exit. IMS DPROP provides the exit code that translates the DB2 information into an IMS update.

    Once again, this is a two-phase commit process.

What environments are supported for synchronous propagation?

Both forward and reverse synchronous propagation are supported **only** in an IMS TM or IMS batch environment, where the IMS subsystem is attached to the DB2 subsystem on the same MVS image.

In reverse, the update to DB2 must come from an IMS transaction or IMS batch job. This is a critically important piece of information that is often misunderstood. It is easier to remember if you keep in mind that DB2 will be signalling BACK to IMS to say that there is a DB2 update to be translated. How can DB2 signal back to IMS if the call did not come from IMS to begin with?

For synchronous, no CICS transaction manager support exists, not even using DBCTL. This is due to the two-phase commit implementation, as architected in IMS. IMS DPROP is using what is available through IMS and DB2 architected interfaces.

For version levels that are needed, see Chapter 7, "General Requirements," on page 37.

What are the performance characteristics of synchronous propagation? What effect will it have on my IMS system?

The main cost of IMS to DB2 synchronous propagation is the cost of the DB2 update itself. The IMS DPROP code is very efficient, is coded in assembler, invokes static SQL calls, and uses VLF as a caching mechanism. All of this amounts to negligible overhead on top of the SQL call.

We have no performance numbers for reverse.

What are the operational implications of using synchronous propagation?

Applications do not have to be altered, but procedures will need to be changed so that IMS transactions will be able to run in mixed mode.

There are system operational considerations, such as making sure that DB2 is up at the same times that IMS is up.

It is possible to change the status of the IMS DPROP system so that IMS transactions can go on, even without DB2 being up. However, this would necessitate a resynchronization prior to going on with synchronized updates. We provide a utility to facilitate a resynchronization.

Where do I specify that I want the IMS Change Data Capture exit or log processing? What is the level of granularity?

> The option is specified on the IMS DBD definition. IMS only allows this at the DBD level or segment level, not at the field level.

> Within IMS DPROP, there are ways to get finer granularity. For example, with LOG-ASYNC processing, there is an option to avoid unnecessary DB2 updates by having the Receiver compare before and after images to see if selected fields have been updated. This could be very useful in a case where the probability that an update to IMS will not affect propagated fields is high.

The SLDS can be very large and can reside on tape. Do these have to be the input to the Selector?

> No, there is a recommended alternative. During archive, split off just those records needed for propagation into a separate data set (called a CDCDS), and (using the sample provided) record this data set on the DBRC recon data set. The Selector has logic to associate an SLDS with a CDCDS, and then use the CDCDS in preference to the SLDS. This is a good performance option for the Selector.

Are there IMS database segments that do not lend themselves to data propagation?

> Yes. IMS segments which have no uniqueness factor within the segment data are not suitable for change data propagation. Some IMS application recognize segments by their position only. For example, the fifth segment of 12 might be recognized as the segment that contains data that specifies the month of May.

> Such positional information will not be contained in 9904 log records, nor will it be available to the IMS CDC exit. Segments of this nature are not suitable for changed data propagation unless they are changed to encode the positional information within the segment data itself.

Assume that SEGB has fields KEYB, B and a block of fields C1 thru Cn. Depending on the value of field B, the C block could be defined as either 4 fields of length 24 or 6 fields of length 16 or 8 fields with varying lengths totalling 96 bytes in all. Propagation of this segment would be directed to 3 different DB2 tables depending on the value of field B. How do I do it?

> This is possible using the standard mapping cases with a WHERE clause on each of the Propagation Requests to examine the value of field B.

Consider the following problem in IMS DPROP synchronous propagation from IMS to DB2: After any propagation error, like an illegal date or non-numeric data existing in IMS, the updating on-line program abends and the transaction is re-queueing automatically to the IMS TM queue. This situation causes a loop because the same error occurs again. Until the error situation is fixed, we need to bypass this re-queueing, because otherwise, the spool becomes full and that means trouble. What solution can we implement without making any changes to the original IMS updating program?

> These situations are good candidates for IMS DPROP "field exit routines". Utilizing field exits the problem IMS data fields could be rendered valid prior to propagation to DB2, cleansing the data in the process. The exit routines can be written in Assembler, Cobol, PL/I, or C (Exit routines can also be written in COBOL II, as long as the conditions listed in an earlier question are met.). Field exits are the most straightforward exit type to implement and are described in detail in the *Customization Guide*. The sample exits are shown in the book and are also shipped in softcopy with the product in

each of the supported languages. Utilizing this method will allow you to proceed without any changes to the IMS updating program.

Chapter 7. General Requirements

This chapter describes the software and hardware requirements of IMS DPROP. It also describes the optional software.

For a complete listing of the requirements and prerequisites, see the *IMS DPROP V3R1 Program Directory*.

The following topics provide additional information:
- "Software Requirements"
- "Optional Software" on page 38
- "Hardware Requirements" on page 39
- "Ordering Information" on page 39

Software Requirements

For planning purposes, the following sections list the software requirements for IMS DPROP. Refer to the *Program Directory for IMS DPROP Version 3.1* for the most current list.

Software Requirements for the IMS DPROP Environment

The following program products are required in the IMS DPROP environment during or after installation:
- z/OS Version 1 Release 6 (5694–A01)
- The following basic or optional features of z/OS Version 1 Release 6:
  - JES2 OR JES3
  - ISPF
  - ISPF/PDF
  - TSO/E
  - SMP/E
  - DFSORT™
- IBM High Level Assembler Version 1 Release 2 (5696-234)

Software Requirements for MQ-ASYNC Propagation

The following program products are required for MQ-ASYNC propagation, IMS to DB2 and IMS to IMS on MVS:
- IMS/ESA® Database Manager Version 6, with APAR PQ51898 (5655-158), or IMS Database Manager Version 7, with APAR PQ51895 (5655-B01)
- IMS/ESA Transaction Manager Version 6, with APAR PQ51898 (5655-158), or IMS Transaction Manager Version 7, with APAR PQ51895 (5655-B01)

  IMS Transaction Manager is needed for online processing only if propagating in message processing program (MPP), IMS Fast Path (IFP), or non-Database Control-Batch Message Processing (non-DBCTL-BMP) regions.
- DB2 Version 7 or higher (5675-DB2)
- MQSeries Version 5 Release 3 or higher (5655-F10)

Software Requirements for Synchronous Propagation

The following program products are required for synchronous propagation, IMS to DB2 on MVS:

- IMS/ESA Transaction Manager Version 6, with APAR PQ51898 (5655-158) or IMS Transaction Manager Version 7, with APAR PQ51895 (5655-B01)

  IMS Transaction Manager is needed for online processing only if propagating in message processing program (MPP), IBM function program (IFP), or non-Database Control-Batch Message Processing (DBCTL-BMP) regions.
- DB2 Version 7 or higher (5675–DB2)

The following program products are required for synchronous propagation, DB2 on MVS to IMS:
- IMS/ESA Database Manager Version 6, with APAR PQ51898 (5655-158) or IMS Database Manager Version 7, with APAR PQ51895 (5655-B01)
- IMS/ESA Transaction Manager Version 6, with APAR PQ51898 (5655-158) or IMS Transaction Manager Version 7, with APAR PQ51895 (5655-B01)

  IMS Transaction Manager is needed for online processing only if propagating in MPP, IFP, or non-DBCTL BMP regions.

Software Requirements for LOG-ASYNC Propagation

The following program products are required for LOG-ASYNC propagation, IMS to DB2 on MVS:
- DB2 Version 7 or higher (5675–DB2)
- IMS/ESA Database Manager Version 6, with APAR PQ51898 (5655-158) or IMS Database Manager Version 7, with APAR PQ51895 (5655-B01)

Optional Software

The following program products are optional in the IMS DPROP environment.
- DataRefresher Version 1 (5696-703) or higher, if you want to use the DataRefresher Map Capture exit routine provided by IMS DPROP.
- DB2 DataPropagator for OS/390® (5655-E60) Version 7, if you want to extend IMS data propagation to the DB2 Universal Database™ for Windows NT, DB2 for AIX, or DB2 UDB for AS/400®. You also need DataRefresher Version 1 (5696-703) or higher, if you want a method of initially populating DataPropagator Relational's data staging area.
- Query Management Facility for OS/390 Version 7, which is a feature of DB2 Version 6 (5645-DB2), if you want to query the IMS DPROP directory tables and audit trail table.
- The Language Environment® Feature of the z/OS Version 1 Release 6 or higher, if you want to write exit routines in high level languages.
- IBM COBOL for OS/390 and VM Version 2 (5648-A25) or higher, if you want to write exit routines in COBOL.
- IBM C/370™ Version 1 Release 2 (5688-216) or higher, if you want to write exit routines in C.
- IBM Programming Language (PL/I) 370 Version 1 Release 1 (5688-235) or higher, if you want to write exit routines in PL/I.
- Customer Information Control System, CICS Transaction Server for OS/390 Version 1 (5655-147) or higher, if you want to support CICS/DBCTL for LOG-ASYNC propagation, IMS to DB2, on MVS. CICS is required only if IMS data is being updated through CICS transactions.

Hardware Requirements

IMS DPROP operates on any IBM processor supported by z/OS Version 1, Release 6, or higher. No special machine configurations are required.

Most IMS DPROP modules are loaded above the 16 megabyte (MB) line.

Ordering Information

To order IMS DPROP or to ask questions about IMS DPROP, contact IBM Direct, your local IBM representative, or your IBM Business Partner.

Chapter 8. IMS DPROP Library

This book is part of a library that supports IMS DPROP. The IMS DPROP library is task-oriented. Each book in the library helps you perform a task, for example, installing IMS DPROP or diagnosing problems.

The following topics provide additional information:
- "Documentation for Installing IMS DPROP"
- "Documentation for Administering IMS DPROP"
- "Documentation for Customizing IMS DPROP"
- "Documentation for Diagnosing IMS DPROP Problems" on page 42
- "Books List" on page 42

Documentation for Installing IMS DPROP

The *Installation Guide* describes the task of installing IMS DPROP.

It also describes how to prepare the MVS environment for IMS DPROP, how to install IMS DPROP and verify proper installation, and how to complete post-installation tasks.

The *Installation Guide* assumes you understand basic IMS DPROP concepts and the IMS DPROP environment, as explained in this book.

Documentation for Administering IMS DPROP

The three *Administrator's Guides* describe the task of administering IMS DPROP. There is an Administrator's Guide for each mode of propagation:
- *Administrators Guide for MQSeries Asynchronous Propagation*
- *Administrators Guide for Log Asynchronous Propagation*
- *Administrators Guide for Synchronous Propagation*

The main administrative tasks for IMS DPROP are divided into four phases.
- Mapping and design—determining the data you are propagating
- Setup—ensuring that your IMS and DB2 systems are ready for propagation
- Propagation—beginning and refining propagation
- Maintenance and control—periodically checking for data consistency or adjust propagation requests

The *Reference* also supports the task of administering IMS DPROP by describing the JCL, the ISPF panels, the utilities, and the control statements associated with the utilities that you can use to propagate data.

Documentation for Customizing IMS DPROP

The *Customization Guide* describes the task of customizing IMS DPROP.

It provides information on how to write exit routines for your IMS DPROP system. This book contains sample segment, field, and propagation exit routines that you can use. It also describes how to design and develop programs that implement your own user asynchronous propagation.

Documentation for Diagnosing IMS DPROP Problems

The task of diagnosing problems is described by *Diagnosis*.

This book contains procedures that help you develop a description of a problem using keywords. The string of keywords you develop can be used to search the IBM Software Support Facility (SSF) or the Early Warning System (EWS) for a similar failure. If you find a similar failure (identified by the same keywords), in one of these databases, you can retrieve and apply the programming temporary fix (PTF).

The *Reference* supports the task of diagnosing problems by describing the audit trail table, which tracks significant events related to propagation.

The *Messages and Codes* contains the descriptions of IMS DPROP messages and codes. They are used in a variety of ways in the diagnostic process. Often, a message is one of the keywords that is used to describe a failure.

Books List

Table 2 is a list of the books in the IMS DPROP library that is organized by task.

Table 2. List of IMS DPROP Books

Task	Book Title	Order Number
Evaluating	• An Introduction • IMS DataPropagator for z/OS: Concepts	• GC27–1211 • SC27–1544
Installing	Installation Guide	GC27–1212
Administering	• Administrators Guide for MQSeries Asynchronous Propagation	• SC27–1217
	• Administrators Guide for Log Asynchronous Propagation	• SC27–1216
	• Administrators Guide for Synchronous Propagation	• SC27–1215
	• Reference	• SC27–1210
Customizing	Customization Guide	SC27–1214
Diagnosing	• Diagnosis • Messages and Codes • Reference	• GC27–1209 • GC27–1213 • SC27–1210

If you want to know more about the other Data Replication products, Table 3 is a list of introductory books for each product.

Table 3. List of Data Replication Introductory Books

Book	Order Number
DataRefresher Version 1 An Introduction	GH19-6993
IBM DB2 Universal Database Replication Guide and Reference	SC26-9920

Notices

This information was developed for products and services offered in the U.S.A. IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs

and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J74/G4
555 Bailey Avenue
P.O. Box 49023
San Jose, CA 95161-9023
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute

these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

The following terms are trademarks of the IBM Corporation in the United States or other countries or both:

AIX	IMS/ESA
BookManager	Informix
C/370	Language Environment
CICS	MQSeries
Database 2	MVS
DataJoiner	MVS/DFP
DataPropagator	MVS/ESA
DataRefresher	MVS/XA
DB2	OS/400
DB2 Universal Database	QMF
Distributed Relational Database Architecture	RACF
DFSORT	RUP
DRDA	S/370
DXT	System/390
IBM	VTAM
IMS	z/OS

Microsoft, Windows®, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

IBM ®

Program Number: 5655-E52

Printed in USA