**170-WP-017-001**

# Catalogue Interoperability Protocol (CIP) - Technical Note on Local Attributes

**White Paper**

**August 1995**

Prepared Under Contract NAS5-60000


**RESPONSIBLE ENGINEER**

Janet F. Hylton Date
EOSDIS Core System Project

**SUBMITTED BY**

George Percivall Date
EOSDIS Core System Project



Hughes Information Technology Systems
Upper Marlboro, Maryland

This page intentionally left blank.

# Contents

## 6. Scenario



## 7. Benefits

## List of Figures



## List of Tables



## Appendix A.

# 1. Introduction

## 1.1 Purpose

The CIP URD contains requirements for CIP B to support “local” attributes for a collection at the local site. The current CIP B Spec v 2.2 presents a method of handling “local” attributes by defining a new schema which includes the CIP attributes in addition to the “local attributes”. These schema are then described in the EXPLAIN database . Any query which is directed at one of these “local” schemas can use the local attributes in addition to the CIP attribute set. The schema can be used by any number of “collections” so the attribute definitions can have a scope from one collection to the entire local site.

There have been issues with the approach described in the CIP Spec v2.2. The purpose of this TN is to identify the issues and alternative solutions for the CIP Spec Consolidation efforts.

## 1.2 Organization

Section 2 identifies the issues surrounding Local Attributes in CIP. Section 3 of this note contains a discussion of Local Attributes in terms of data and services. Section 4 of this document addressses the proposed solutions. Section 5 explores the suggested approach, Section 6 provides a scenario which illustrates the interaction between and among the ICS components. Lastly, section 7 provides an impact assessment on the existing CIP/ICS documentation and Appendix A provides additional background material.

## 1.3 Review and Approval

This document was prepared as part of the ECS Contract between NASA–GSFC–ESDIS and Hughes as described in ECS Engineering Support Directive #25, ECS Extensions Support.

Questions regarding technical information contained within this Paper should be addressed to the following ECS and/or GSFC contacts:

- ECS Contacts
  - Janet F. Hylton, (301) 925-0466, janet@eos.hitc.com
- GSFC Contacts
  - Yonsook Enloe, (301) 286-0794, yonsook.enloe@gsfc.nasa.gov

Questions concerning distribution or control of this document should be addressed to:

Data Management Office
The ECS Project Office
Hughes Information Technology Systems
1616 McCormick Drive
Upper Marlboro, Maryland 20774-5372

This page intentionally left blank.

# 2. Issues

This section provides as high level view of the issues that have been identified and the potential alternative implementations.

## 2.1 Issues analysis

This analysis focuses on

Identifying the major issues;

Defining the issue;

Providing alternatives for each issue and;

Recommending an alternative.

Collectively this information served to determine the recommended approach. The following describes each of the issues in the analysis terms identified above.

### 2.1.1 Are "local" attributes required in CIP Release B?

The issue of whether searchable "local attributes" are needed for CIP B has been raised based on the experiences of the CIP Demonstrator team. None of the datasets included in the CIP Demonstrator had searchable attributes that were not covered by the standard CIP attribute set.

The consensus of the Local Attribute TN team was that local attributes were needed by many datasets and the Demonstrator experience was not a sufficient sample. EOSDIS has discovered a large number of local attributes are required by many of its supported datasets. An Extract of the ECS local attributes for MODIS is included as Appendix A of this document.

It was also pointed out that the ability to do inventory searches based on local attributes is a major distinguishing feature between CIP and earlier interoperability oriented systems such as GEO and IMS.

It is important to notice that the IMS will soon be able to support local attribute searches. A description of the IMS messages to accomplish this are included as Appendix A of this document and can be found on the web at: http://harp.gsfc.nasa.gov/documents/extn/extended_search.html

In conclusion the overall consensus was that local attributes should be supported in CIP Release B.

### 2.1.2 Scope of queries containing "local" attributes

Before an implementation of "local attributes" is defined, it is necessary to identify and define the scope of a query containing "local attributes". Some of the alternatives include:

Provider Archive Collection  Product Descriptor search only

Provider Collections (Archive and Theme) Product Descriptor search only

Product descriptor and collection searches

Local searches only (within single site)

### 2.1.3 Characteristics of Local Attributes

It is believed that in order to provide a workable design it is first necessary to understand the data that the design will need to support. Given that this information for local attributes is not readily available within the EO community it was first necessary to attempt to locate any previous analysis and from this information project a reasonable strategy for CIP. The following presents the results of this fact finding mission and assumptions that have been made for CIP.

The ECS data analysts have conducted an extensive exploratory effort to determine the types and numbers of local attributes that are anticipated for ECS. A subset of the results of this effort are captured in Appendix A. To summarize, it appears that local attributes tend to fall within several information categories; Science Specific, Calibration and Quality Assessment. These attributes may also appear across several collections in a discipline.

The ECS MODIS data providers have identified approximately 150 unique Local Attributes and the ECS CERES data providers have identified approximately 20. Therefore it is reasonable to assume that the numbers of local attributes will vary depending on each individual data providers needs and how these needs are currently be supported within the current metadata specification.

### 2.1.4 Implementation Difficulty

The current strategy outlined in the CIP Specification requires that the RMA become knowledgeable about the inter workings of the Explain Environment. This level of knowledge was further characterized in the first draft release of the Collections Manual in May 1997. After carefully evaluating the level of expertise required to support the definition of the Local Attributes in the Explain it was determined that an alternate strategy should be investigated.

# 3. What are Local Attributes?

The Local Attributes are characteristics about collections which cannot be captured within the current specified metadata for CIP/ICS. They are attributes which are unique to a discipline or to a specific scientific investigation. These attributes assist the user in narrowing his search to only those products which have the potential of providing the needed information. Therefore, they should be both searchable and retrievable. The following presents an overview of the data and further the anticipated services that will be required to support local attributes.

## 3.1 Data

The Local Attribute Definition is the core concept behind the local attributes. This definition specifies the characteristics of the local attributes. It includes the name, meaning, units of measure and range of values. There are Local Attribute Values which are captured at the product level such as Average Focal Plane Temperature Local Attribute whose range of valid values are between 0.0 and 400.0. The actual value, i.e. 200.2, would be expressed at the product level while the attribute name, meaning, units of measure, and range of values, would be at the collection level.

## 3.2 Services

Because of the nature and usage of the Local Attributes, both the search and present services should be supported within CIP. The following describes each of these services.

### 3.2.1 Search

Searching Local Attributes should be supported for Provider Archive and Product. For the Provider Archive the Attribute Name should be searchable. For the Product the Attribute Value should be searchable. Additionally, searching Remote Local Attributes should also be supported. Reference Section 6.2 for a discussion on searching Remote Local Attributes.

### 3.2.2 Present

The present service will allow the Provider Archive local attributes and their associated meanings, units of measure, and range of values, to be returned as a result of a Provider Archive collection query. The product query for local attributes will return local attribute values. The actually composition of the present is addressed in section 5 of this TN.

This page intentionally left blank.

# 4. Proposed Solutions

The following identifies and briefly describes each of the proposed solutions. Additionally, although there are many advantages and disadvantages for each proposal, only the major concerns are described for each solution.

## 4.1 Using the Explain to capture Local Attributes

This solution is the approach presented in the CIP Spec. It relies on the idea that the Local Attributes are identified and described in the Explain Database. The details surrounding this solution are contained in Appendix A.

The major advantage of this approach is that the current CIP design supports all data definitions in the Explain Database which is designed to accommodate the identification and description of all CIP attributes. Additionally, the Explain provides the values for the local attributes to the Client.

The major disadvantage is the complexity associated with maintaining the integrity of the Local Attribute Definition in the Explain environment. This maintenance activity is also described in Appendix A.

## 4.2 Using the Collection Descriptor to capture Local Attributes

This solution which is described in detail in Appendix A is an alternative to the Explain. It employs the use of the Collection Descriptor to explain the Local Attributes rather than the Explain Database itself.

The major advantage is that the complexity associated with creating and maintaining the Explain is minimized with this approach.

The major disadvantage is that the existence of Local Attributes will only be know to the local database. The RM (Explain), which provides search information to the Client, will not have the needed information to readily determine if a collection of interest contains local attributes prior to accessing the Collections Descriptor

## 4.3 Using a Hybrid combination of the previous two proposals

This approach suggests that the information in the Explain and Collection Descriptor in the local database work in harmony to provide the Local Attribute information to the client. This solution is fully defined in sections 5 and 6 of this note

The major advantage is that the explain maintenance activity will be limited to identifying the existence of Local Attributes for a given collection thus limiting the need for the RMA to directly interact on a maintenance level with Explain.

The major disadvantage is that the Explain will not be used to capture the semantics of each specific Local Attribute. They will be captured in the local database. This will require that the RM obtain specific information, i.e., name, meaning, and range of values, regarding the local attributes from the database.

# 5. Suggested Approach

The Hybrid Solution which proposes a combination of the Explain solution and the Collection Descriptor is the suggested approach for representing Local Attributes in CIP.

## 5.1 Local Attribute in Explain

The following identifies and describes how local attributes will be managed by the Explain Database.

**Include the new Local Attributes in the Collection Descriptor Schema**
The following are the attributes for the Local Attribute definition. It is anticipated that these attributes will be represented as additional elements for the Collection Schema in the Explain.

| | | | |
|---|---|---|---|
| | 0{ AttributeDef }n | | |
| **4222** | **AttributeDef** | **Compound** | |
| 4224 | AttrName | WordList | UseAttribute |
| 4226 | Meaning | String | |
| 4227 | (UnitsofMeasurement) | String | |
| 4228 | (Range of Values) | String | |

**Include the following attributes in the Product Descriptor Schema**

| | | | |
|---|---|---|---|
| | 0{AttrValue}n | | |
| **4221** | **Attr Value** | **Compound** | |
| 4232 | LocalAttributeValue | String | UseAttribute |
| 4224 | AttrName | WordList | Use Attribute |

The LocalAttributeValue will contain the value associated with the attribute name specified in the collection and in the product.

**Include Associated db in the Database Info record.**
The thinking here is that for each databaseinfo record, which is equivalent to a collection in Explain, that contains local attributes, an Associated db attribute would be specified. The contents of the Associated db attribute (Local Attribute DB) would serve several useful purposes. First it would allow the RM to determine from Explain if local attributes exist for the collection. This information would then allow the RM to construct the appropriate query structure.
Second, as specified in the protocol, it also allows a query to target multiple db's which logically could be interpreted as accessing multiple tables in the local database.

**Identify a new Element Set Name (present view) to incorporate Local attribute**
This may include several views to include

Collection “Brief” + Local(Attribute Name + Meaning + Unite of Measurement + Range of Values)
Local only (Attribute Name + Meaning + Units of Measurement + Range of Values)
Collection “Full” + “Local”
Product “Brief” + “Local”
Product “Full” + “Local”

Please note that these are only views and Element Set Types would need to be identified.

## 5.2 Local Attribute Definition in the Database

The Local Attribute Definition in the database will mirror the schema represented in the Explain. That is, each attribute identified in the schema will represent columns in a relational table. The values for the attributes will be contained in each row. Since the actual Local Attribute will be represented as a value for Attribute Name it is not necessary to identify tag values for each of the local attributes unless the chosen implementation strategy dictates their existence. Nor is it necessary to identify and define any of the specific characteristics as required for Explain elements such as datatypes, maximum size, minimum size, etc. These characteristics are identified for the AttrName and will be described such that any value can be accommodated. the following tables provide an example of the Collection Descriptor Table and the Local Attribute Table respectively

***Table 5-1. Collection Descriptor Table***

| ItemDescriptorID | ItemDescriptor Name | Purpose | Abstract | Processing Level ID | Archive Center ID | Processing Level Descriptor | --- |
|---|---|---|---|---|---|---|---|
| CID-modis001 | | | | | | | |
| | | | | | | | |

***Table 5-2. Local Attribute Table***

| ItemDescriptorID | Attribute Name | Meaning | Units of Measurement | Range of Values |
|---|---|---|---|---|
| CID-modis001 | Percentchangedpixels | Percentage of pixels which were flagged as land coverage change | percentage | 0-100 |
| | | | | |

# 6. Scenario

## 6.1 Operational Scenario Local Search

The following is an operational scenario which illustrates the logical sequence of activities that will occur when the User targets Local Attributes in a local search request. The objectives of this scenario are to illustrate and describe the interaction between the Client, Retrieval Manager, and overall Database Environment. This scenario assumes the following:

1. The user was previously provided a list of collections as a result of a collection query
2. The selected collection in step 2 is a local provider archive collection.
3. Product Descriptors exist in the ICS database.

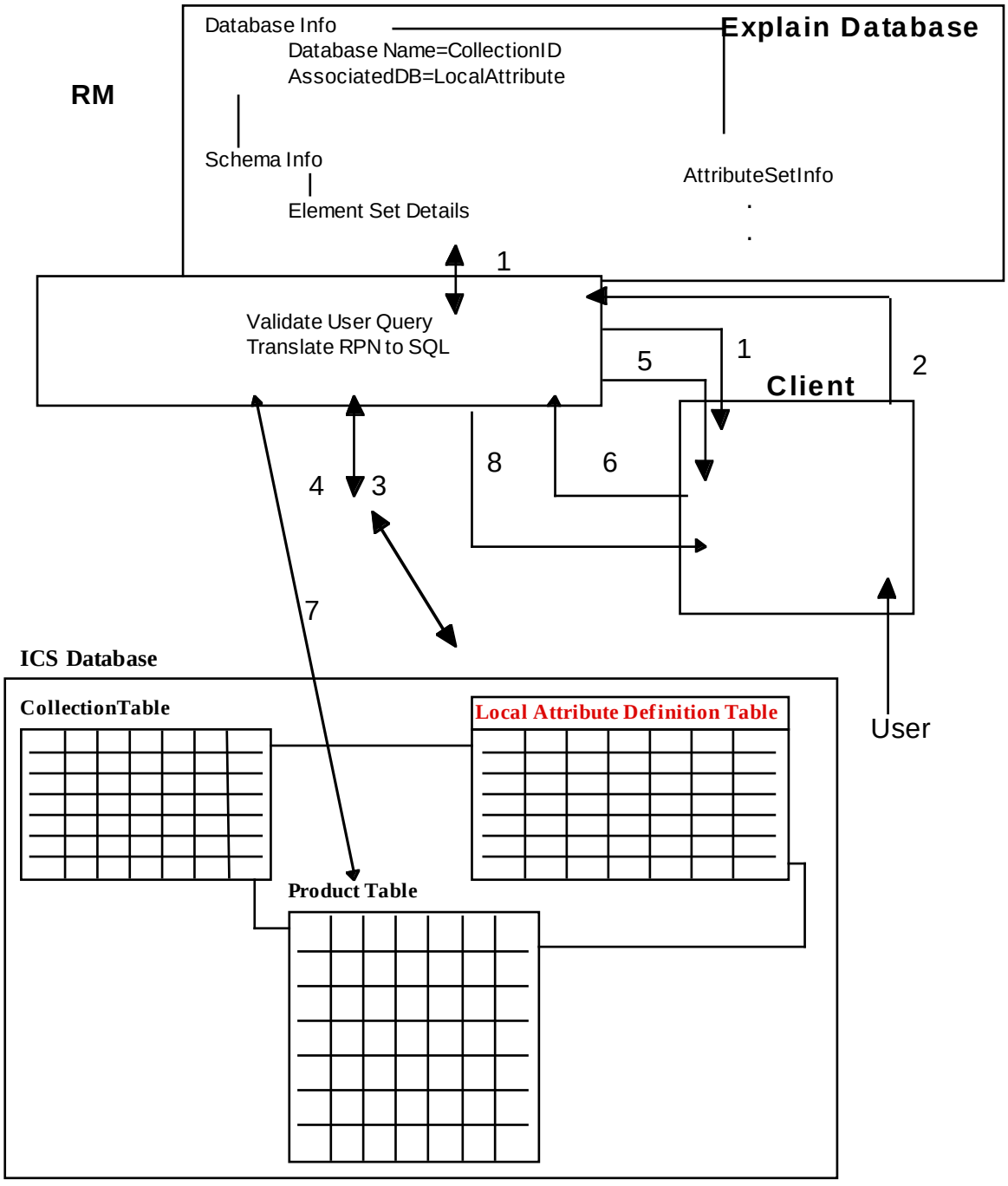



***Figure 6-1. Operational Scenario Local Search***

**Step1** Results from a previous Collection Search request have been presented to the user.

**Step2** User request list of all Local Attributes associated with a specified / selected collection, Client forwards the request to the RM.

**Step3** RM validates the request and determines if the requested Coll has AssociatedDB, If the AssociatedDB exist for the selected collection then the RM translates the RPN to SQL, which targets the Local Attribute Definition Table for the specified collection.

**Step4** The results (AttrName, Meaning, RangeofValues, and UnitsofMeasurement) are sent back to the RM.

**Step5** The RM presents the results to the User via the Client

**Step6** The user then request from the list of local attributes all product descriptors for the selected collection and local attributes whose attribute values = x.

**Step7** The RM validates the request, translates the RPN to SQL which targets the Product Table. The results are then sent back to the RM.

**Step8** The RM presents the results to the User via Client.

## 6.2 Operational Scenario Remote Search

The following illustrates the sequence of activities that will occur when the user targets Local Attributes in a remote search request. This scenario assumes the following:

1. The user was previously provided a list of collections as a result of a collection query
2. The selected collection in step 2 is a remote provider archive collection.
3. Product Descriptors exist in the remote ICS database.

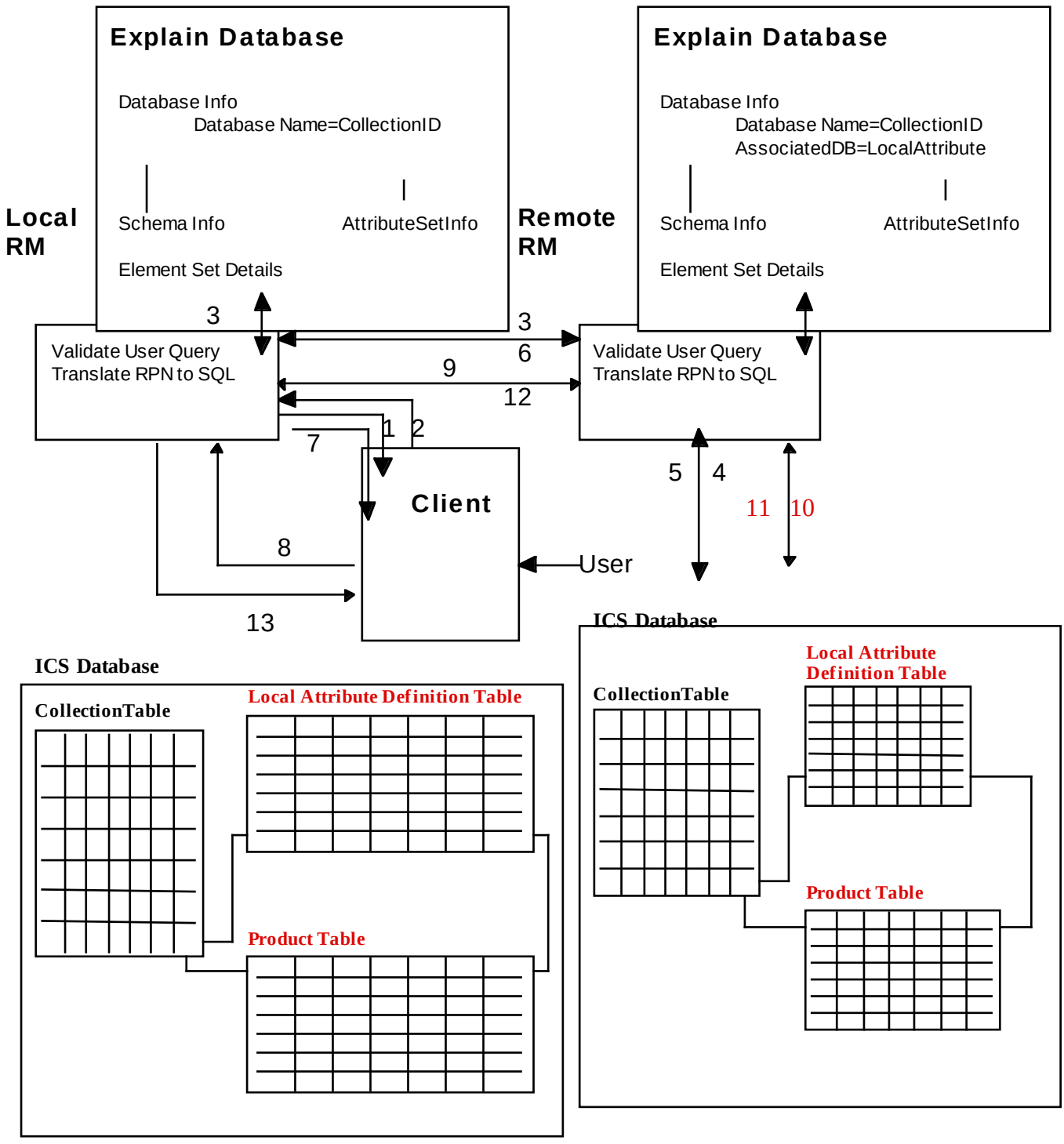



***Figure 6-2. Operational Scenario Remote Search***

**Step1** Results from a previous Collection Search request have been presented to the user.

**Step2** User selects a Provider Theme Collection which includes Remote Provider Archive Collections and requests all local attributes for a Remote Provider Archive Collection be presented if available. Client forwards the request to the RM.

**Step3** The RM validates the request and submits the RPN to the remote RM.

**Step4** The remote RM validates the request and determines if the requested Coll has AssociatedDB, If the AssociatedDB exist for the selected collection then the RM translates the RPN to SQL, which targets the Local Attribute Definition Table for the specified collection.

**Step5** The results (AttrName, Meaning, Range of Values, and Units of Measurement) are sent back to the RM.

**Step6** The remote RM forwards the results to the requesting RM

**Step7** The RM presents the results to the User via the Client

**Step8** The user then request from the list of local attributes all product descriptors for the selected collection and local attribute whose attribute value = x. The Client forwards the request to the RM.

**Step9** The RM submits the RPN to the remote RM.

**Step10** The remote RM validates the request and translates the RPN to SQL which targets the Product Table.

**Step11** The results are sent back to the RM.

**Step12** The remote RM submits the results to the requesting RM.

**Step13** The RM presents the results to the user via the Client.

## 6.3 RMA Maintenance Scenario

The following briefly describes the maintenance activities associated with creating the Local Attribute information in the Explain and the local database (Local Attribute Definition Table).

The RMA receives a request to include a Local Attribute SST for the MODIS001 Collection.

The RMA determines from the Explain that the specified collection does not contain Local Attributes.

The RMA includes the attribute “AssociatedDB” and attribute value “Local Attribute” in the Database Info for the MODIS collection.

The values associated with the attribute name=SST, Range of Values, Meaning, and Units of Measurement , will then be included in the Local Attribute Definition Table.

This page intentionally left blank.

# 7. Benefits

The following addresses some of the benefits that may be realized as a result of implementing the above approach.

User can target a remote search at a Provider Archive Collection and associated remote Local Attributes.

The RMA would simply need to update the db info, associated db attribute for a collection to reflect that local attributes for that collection exist.
Update the Local Attribute table to include the new local attribute name and value if required (may already exist). Build the association between the Local Attribute entry and the Collection.

Increases the potential for interoperability with other systems, i.e. GEO.

This page intentionally left blank.

# Appendix A.

## A.1 Local Attribute Requirements

**The following are extracts from the CIP URD**

**UR Id : 50**

**Source : DPRS TN [R1, Section 4.1.6]**

**CIP-BS Team**

**Priority : 1**

**Need : A**

**Qualifier : RP**

**The CIP shall support 'local' sets of additional attributes to the standard set. The CIP shall not reject such 'local' attributes but forward them with the recognised standard attributes.**

**Note : These 'local' sets of attributes will not be defined in the CIP specification.**

**However, the CIP will support the access and retrieval of those 'local' attributes**

**by remote catalogue systems Local attributes are likely to be used by a particular**

**data provider and not available throughout the CIP domain.**

**UR Id : 51**

**Source : URD 2.0 Review [NASA/LR/01]**

**Priority : 1**

**Need : B**

**Qualifier : RP**

**The CIP shall support 'collection specific' sets of local attributes as extensions to the standard set. The CIP shall not reject such 'collection specific' attributes but forward them with the recognised standard attributes.**

**Note : These collection specific sets of attributes will enable the user to request the**

**particular attribute set which is specific for searching the required collection**

## A.2 Current Implementations

**V0**

## A.3 Current Solution (Rel B Spec) using Explain

### Creating a New Entry - Present Service

**The assumptions for this option are that the attribute set, attribute type, attribute value and attribute default will include CIP standard specification as reference in the CIP Release B Specification. This standard specification will be used with the Search Service for the collection/product. Therefore only the following Explain Classes will require an entry:**

Tag Set Info

Schema Info

Element Set Detail (only if Local Brief Required)

Per Element Set Details (Local Elements Only)

Database Info

**Additionally the following relationships will need to be established:**

Database Info to Attribute Set Info

Database Info to Schema Info

Database Info to Query Type

Database Info to Diagnostic Sets

Database Info to Record Syntax Info

Schema Info to Element Set Details

Element Set Details to Per Element Set Details

Tag Set Info to Schema Info

Tag Set Info to Per Element Set Details

### A.3-1 Creating a New Entry - Search and Present Service

**This option requires that additional entries beyond the CIP generic entries be made for the Explain search objects. The following identifies the data classes that will require an entry to support both search and present.**

Tag Set Info

Schema Info

Database Info

Element Set Details (Local brief required)

Per Element Set Details (Local Elements)

Attribute Set Info (Local)

Attribute Value (Local Attributes)

Attribute Default

Attribute Type

**Additionally the following relationships should also be established for the new entry.**

Database Info to Attribute Set Info

Database Info to Schema Info

Database Info to Query Type

Database Info to Diagnostic Sets

Database Info to Record Syntax Info

Schema Info to Element Set Details

Element Set Details to Per Element Set Details

Tag Set Info to Schema Info

Tag Set Info to Per Element Set Details

Attribute Set Info to Attribute Type

Attribute Type to Attribute Value

Attribute Value to Attribute Default

Element Set Details to Record Syntax

**The following Figures A-2 - A-4 illustrates the steps required to add a new local collection entry into the Explain Database.**

**Step 1 is required for the Search Service.**

**Step 2 is required for the Present Service**

**Step 3 creates an entry in the Database Info and is required for all new collections.**

**If both the Search and Present Service for the Local Attributes is required then Steps 1 through 3 should be followed when adding the Local Collection. If the Present Service is the only requirement then Steps 2 and 3 need to be followed.**

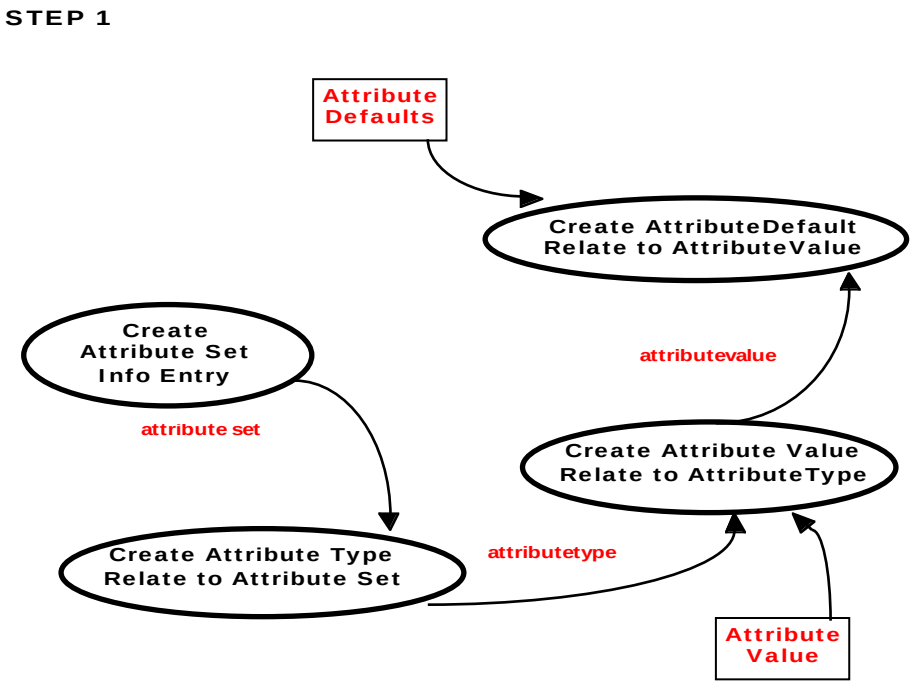



***Figure A-2.. Add New Local Collection Entry to Explain -Search Service***

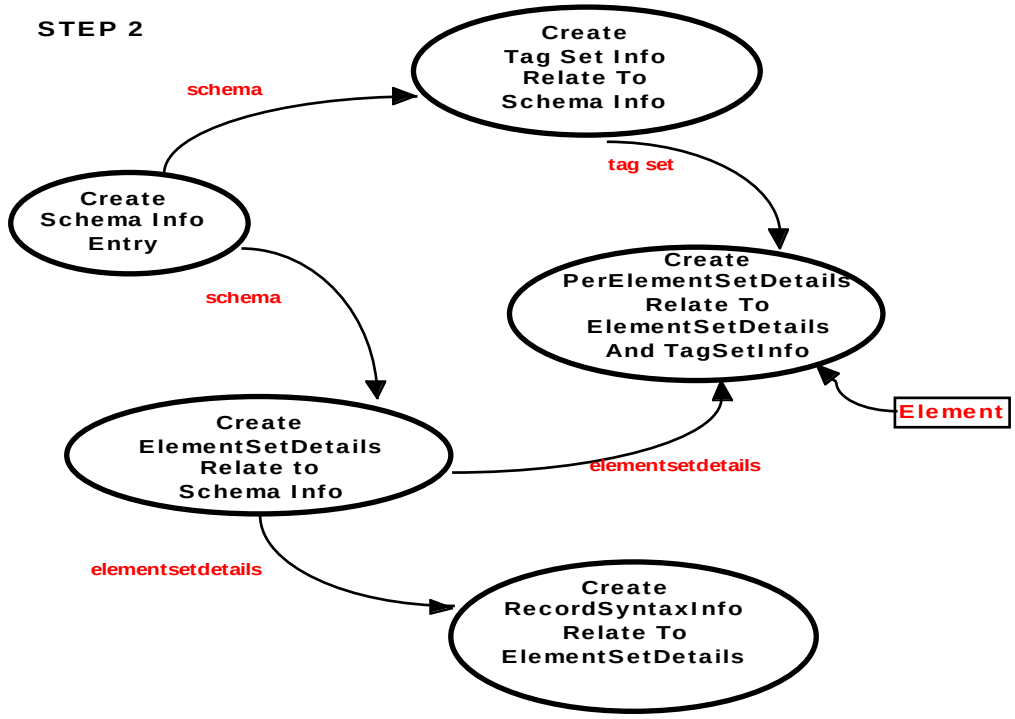



***Figure A-3. Add New Local Collection Entry to Explain -Present Service***

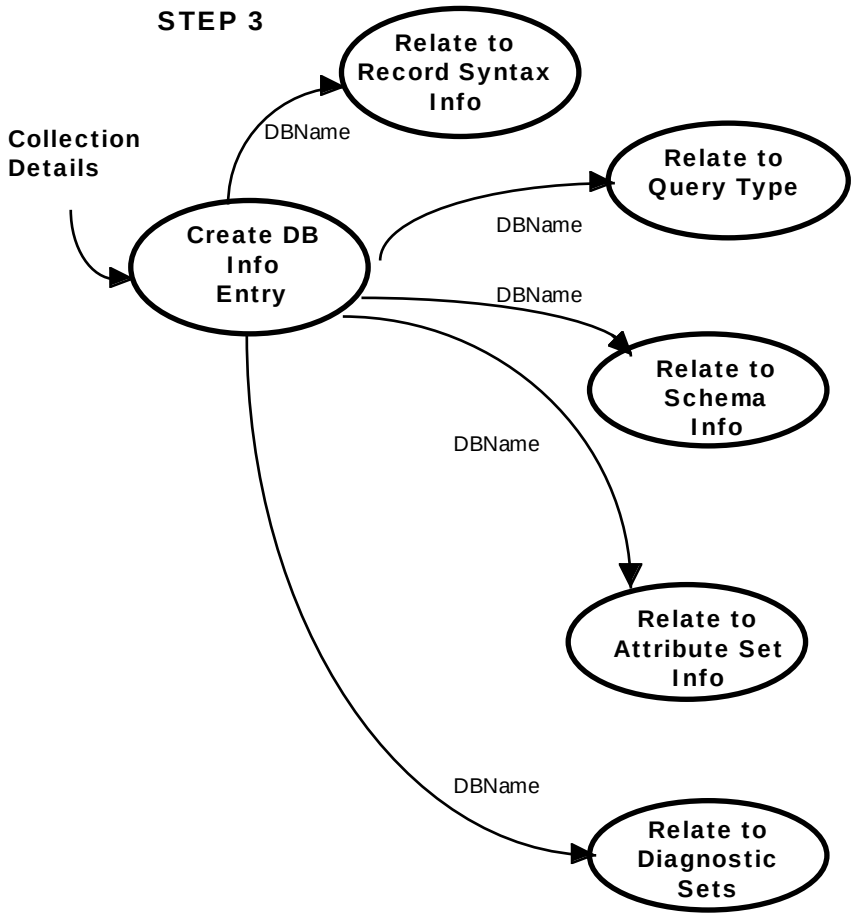



***Figure A-4. Add New Local Collection Entry to Explain***

## A.4 Local attributes in Collection Descriptor

**The following summary was provided by Clive Best:**

Therefore see below as to how Local attributes could be defined and incorporated via the Collection Descriptor ( for Search definition) and via the Product Descriptor (for Retrieval). The following additions to the CIP schema would be needed (additions to Section C.3 of the Spec). In this case Local attributes would be supported as a Flat definition list of Search Targets defined in the Collection Descriptor. In present they would be returned as an ID, value list in a "compound element" in each product descriptor.

A set range of USE attributes within CIP (say 4500 - 4600) would need to be reserved for use as local attribute IDs.

Additions to Collection Descriptor (ARS)

---

| TAG | Element | Value Syntax |
|---|---|---|
| 4221 | ( LocalAttributes ) | Compound |
| 4221 | LocalAttributes | Compound |
| 4222 | 1{ AttributeDef } n | Compound |
| 4222 | AttributeDef | |
| 4223 | UseID | INTEGER |
| 4224 | AttrName | STRING |
| 4225 | StructAttr | INTEGER |
| 4226 | Meaning | STRING |

Additions to Product Descriptor

---

| TAG | Element | Value Syntax |
|---|---|---|
| 4227 | 0{ LocalElements }1 | COMPOUND |
| 4227 | LocalElements | COMPOUND |
| 4228 | 1{ AttributeValue }n | COMPOUND |
| 4230 | AttributeValue | COMPOUND |
| 4231 | ElemID | INTEGER |

4232 ElemValue ( Dynamic - defined above in StructAttr )

## A.5 ECS Local Attributes for MODIS

The following table reflects a very small subset of local attributes that were captured for the ECS MODIS Collections. The ECS term for Local Attribute is PSA (Product Specific Attribute). Therefore to fully understand this table it will be necessary to substitute "Local Attribute" for "PSA".

***Table A.5. ECS Local Attribute Definitions***

| PSA Name | Data Type | Data Length | Collection Short Name | RangeofValues |
|---|---|---|---|---|
| AverageBlackbodyTemperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| AverageFocalPlane1Temperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| AverageFocalPlane2Temperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| AverageFocalPlane3Temperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| AverageFocalPlane4Temperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| AverageMirrorTemperature | float | F(7)(2) | MOD021KM | 0.00-400.00 |
| CirrusCloudDetectedPct_IR | float | F(7)(2) | MOD06_L2 | 0.00 - 100.00 |
| ClearPct250m | float | F(7)(2) | MOD35_L2 | 0.00 - 100.00 |
| CloudCoverFractionPct_VIS | float | F(7)(2) | MOD06_L2 | 0.00 - 100.00 |
| CloudCoverPct250m | float | F(7)(2) | MOD35_L2 | 0.00 - 100.00 |
| CloudPct_IR | float | F(7)(2) | MOD05_L2 | 0.00 - 100.00 |
| CloudPhaseUncertainPct_IR | float | F(7)(2) | MOD06_L2 | 0.00 - 100.00 |

## A.6 V0 IMS Local Attribute Capability

The following was extracted from the VO Extended Search Criteria Web Page. It is being presented in this document to provide insight into the various ways in which existing systems are accommodating local attributes.

Introduction

In the IMS, only search fields that are common across all (or most) datasets are supported. Many data centers would also like to provide searching on additional attributes, which may be common across several datasets, or specific to one. The flexibility of Web-based interfaces should make support for this type of searching easier to implement in the Web Gateway. It is expected that some such specialized search parameters will apply to cross-dataset or cross-data center searches. The planned implementation will provide for extended search criteria as defined by individual data centers, and will allow data centers to coordinate in designing specialized search criteria that apply to some subset of the IMS datasets. Because this implementation calls for the IMS client to construct a custom search screen for each extended search criterion, we will rely on

the SPECIALIZED_CRITERIA group, which can be used to define any of the following types of user input interfaces:

fill-in-the-blank (_____)

fill in a range (from _____ to _____)

select one number from a range

select one item from a list

select one or more items from a list

We plan to support an extended search capability via URL as well, to provide access to specialized search interfaces available at various data centers.

Assumptions

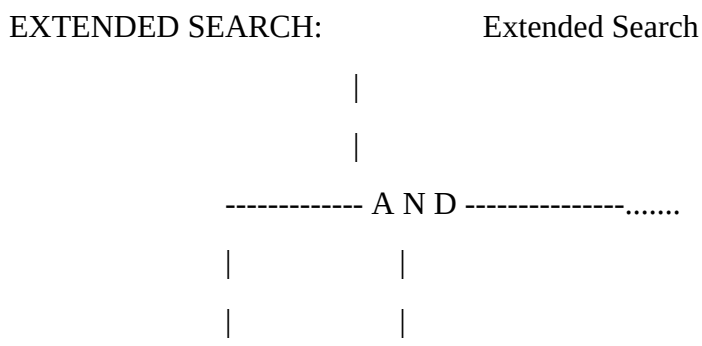
1. Extended Search Criteria will be dependent with the valids. That is, when a user has selected standard search criteria, only the extended search parameters applicable to those datasets matching the selected criteria will be available. Likewise, when a user selects extended search criteria, only those datasets that support the selected extended search criteria will be available. If there are conflicts within the user's final search criteria, the client will notify the user of the compatibilities before the search is sent.

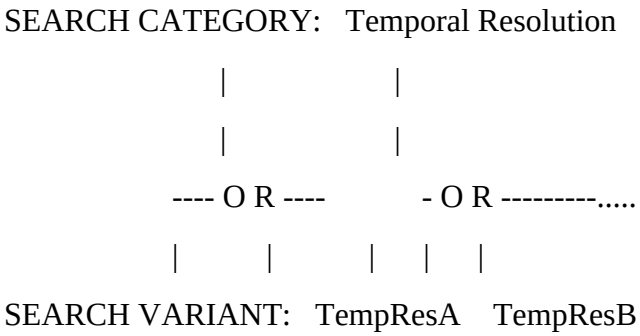
2. Path/Row will be treated as a geographic specification, not an extended search. Search by Granule ID will be implemented as a separate search type. Thus, all extended search criteria will be compatible with geographic search criteria.

3. Similar and related search criteria will be grouped in categories. For example, different data centers may define "Cloud Cover" differently, but all of the variants of cloud cover search will be grouped together.

4. Unique search criteria (i.e., not variants) will be the only members of their categories.

5. In general, extended search criteria categories should be AND'ed (as are the different standard search criteria). However, if a user specifies different variants of a single category, these variants should be OR'ed within the category.






6. For the (rare) case of multi-component extended search criteria, each component should be defined separately, with comments indicating to the user which other components must be specified in conjunction with this one. All should belong to the same category. Individual search components (e.g. START_DATE and STOP_DATE) will be AND'ed. (The data center is responsible for making this determination).

User Scenario

User selects a particular dataset or datasets from the dataset valids list. Once datasets are specified, choices in the extended search list are limited to those specialized search criteria categories (if any) applicable to all the selected datasets. Within each valid extended search category, those variants applicable to any of the selected datasets will be available. For example, the user may select the single dataset-specific search attribute available for the selected dataset(s), say "Percent Cloud Cover".

- OR -

User selects a particular extended search attribute from a "Valids" list. (A search name with a data center ID prefix or suffix indicates that the search is applicable only to dataset(s) at that data center. A search with no prefix or suffix in the name is applicable across two or more data centers.) For example, the user may select "Percent Cloud Cover".

The selection of an extended search attribute limits available choices in DAAC and dataset search fields, as well as other search fields.

- THEN -

A separate button activates a dialog box prompting the user for search criteria associated with the extended search attribute. For example, the user may specify "Percent Cloud Cover from 0 to 20". Search is sent to selected data centers. Results are displayed as usual.

Defining Extended Search Criteria to the Client

Extended search criteria are defined to the Client in a client support file. Extended search definitions are submitted by the DAACs using ODL objects. A SPECIALIZED_CRITERIA group is used to define each extended search.

Examples of Extended Search Definitions

```
GROUP = SPECIALIZED_CRITERIA
  CRITERIA_NAME = "PERCENT CLOUD COVER"
  SEARCH_CATEGORY = "CLOUD COVER"
  CRITERIA_TYPE = "INTEGER"
  COMMENT = ("Description of cloud coverage goes here.",
             "Assume it is listed as a percentage.")
  RANGE = Y
  CRITERIA_MIN = 0
  CRITERIA_MAX = 100
END_GROUP = SPECIALIZED_CRITERIA

GROUP = SPECIALIZED_CRITERIA
  CRITERIA_NAME = "CLOUD COVER/EDC"
  SEARCH_CATEGORY = "CLOUD COVER"
  CRITERIA_TYPE = "STRING"
  COMMENT = ("Description of cloud coverage goes here.",
             "Assume it is listed as a series of values.",
             "User can select one or more of the values.")
  CRITERIA_VALUE = ("0 to 20 percent", "20 to 40 percent",
    "40 to 60 percent", "60 to 80 percent", "80 to 100 percent")
  SELECT_NUM = MANY
END_GROUP = SPECIALIZED_CRITERIA

GROUP = SPECIALIZED_CRITERIA
  CRITERIA_NAME = "TEMPORAL RESOLUTION/MSFC"
  SEARCH_CATEGORY = "TEMPORAL RESOLUTION"
  CRITERIA_TYPE = "STRING"
  COMMENT = "Description of temporal resolution goes",
            " here.  User can chose one of a specified",
```

" set of keywords."

CRITERIA_VALUE = ("1 DAY", "15 MIN. SUMMARY", "5 DAY

AVERAGE", "DAILY TOTAL", "MONTHLY MEAN",

"MONTHLY TOTAL")

SELECT_NUM = ONE

END_GROUP = SPECIALIZED_CRITERIA

Mechanism for Defining Extended Search Criteria

Extended Search criteria will be coordinated by the IMS V0 Client team. Any V0 data center may submit a new Extended Search definition. The Extended Search Coordinator will work with the data center to refine the new definition if needed, and to group it with other related Extended Search criteria. The Extended Search Coordinator may assist in cross-data center negotiations to standardize similar Extended Search Criteria. Once the new search definition is finalized, the Coordinator will add it to the "Extended Search Valids" maintained by the Client, and publish the definition where all data centers can access it. Data centers can include an optional EXTENDED_CRITERIA_AVAIL sequence string parameter in any DATASET group in their valids, listing all defined Extended Searches that apply to that dataset. Data centers should not re-submit the search definition within the dataset valids.

Rules for Defining Extended Search Criteria

A SPECIALIZED_CRITERIA group is used to define each extended search. The CRITERIA_NAME value should be unique, and should include the data center ID unless the particular search has been jointly defined by two or more cooperating data centers. nSPECIALIZED_CRITERIA groups should be constructed according to the rules and examples outlined in the Specialized Criteria document.

Specifying an Extended Search to the Server

Extended search criteria will be specified to the servers in the Inventory Search message. They should be grouped by category so that the query logic is preserved (AND search categories, OR within search categories). That is, multiple variants of an extended search category will be described with multiple SPECIALIZED_CRITERIA groups, all within a single EXTENDED_SEARCH group. Each extended search category should have its own EXTENDED_SEARCH group in the Inventory Search message. In the rare case of a multi-component search, all search components should be included in the same EXTENDED_SEARCH group. The server is expected to recognize which components within the search category are to be AND'ed, and to return a STATUS_CODE_COMMENT indicating an error if all required components are not specified.

Examples

User wants monthly data from one data center.

GROUP = INVENTORY_SEARCH

```
. . .
  DATA_CENTER_ID = "MSFC"
  GROUP = EXTENDED_SEARCH
    GROUP = SPECIALIZED_CRITERIA
      CRITERIA_NAME = "TEMPORAL RESOLUTION/MSFC"
      CRITERIA_VALUE = "MONTHLY MEAN"
    END_GROUP = SPECIALIZED_CRITERIA
  END_GROUP = EXTENDED_SEARCH
. . .
END_GROUP = INVENTORY_SEARCH
```

User wants data with 35% cloud cover or less. For the datasets the user has selected, two definitions in the "CLOUD COVER" category are available, one general (jointly agreed to by two or more data centers), and one data center specific. The user specifies values for both variants of cloud cover, and submits one Inventory Search.

```
GROUP = INVENTORY_SEARCH
. . .
  DATA_CENTER_ID = ("EDC","GSFC")
  GROUP = EXTENDED_SEARCH
    GROUP = SPECIALIZED_CRITERIA
      CRITERIA_NAME = "PERCENT CLOUD COVER"
      CRITERIA_MIN = 0
      CRITERIA_MAX = 35
    END_GROUP = SPECIALIZED_CRITERIA

    GROUP = SPECIALIZED_CRITERIA
      CRITERIA_NAME = "CLOUD COVER/EDC"
      CRITERIA_VALUE = ("0 to 20 percent", "20 to 40 percent")
    END_GROUP = SPECIALIZED_CRITERIA
  END_GROUP = EXTENDED_SEARCH
. . .
```

END_GROUP = INVENTORY_SEARCH

Indicating Which Extended Search Criteria Were Used

The dependent valids mechanism should, in general, keep data centers from getting Extended Searches that they do not support. However, a search message may contain several variants of an extended search category, only one of which applies to any individual dataset.

To indicate which of the search variants was used in generating search results, and EXTENDED_CRITERIA_USED parameter (ODL sequence string) should be included in each DATASET group of the Inventory Results message. The EXTENDED_CRITERIA_USED parameter should list the CRITERIA_NAME of each SPECIALIZED_CRITERIA applied to the dataset.

In the unlikely event that an Extended Search is sent to a data center that does not handle these searches, the absence of the EXTENDED_CRITERIA_USED parameter will allow the Client to filter out these unwanted search results for the user.

Examples

```
GROUP = INVENTORY_RESULT
  ...
GROUP = DATASET
   ...
   EXTENDED_CRITERIA_USED = ("PERCENT CLOUD COVER",
         "PRODUCT MAP PROJECTION")
GROUP = GRANULE
   ...
   END_GROUP = GRANULE
   END_GROUP = DATASET

GROUP = DATASET
   ...
   EXTENDED_CRITERIA_USED = ("CLOUD COVER/EDC")
GROUP = GRANULE
   ...
   END_GROUP = GRANULE
   END_GROUP = DATASET
```

END_GROUP = INVENTORY_RESULT

Indicating Extended Criteria Values in Search Results

Obviously, if the user has searched on a cloud cover range, s/he would like to know the percent cloud cover for each granule returned. Similarly, when searching for data at a specific temporal resolution, the user needs to know the temporal resolution of each dataset. Each GRANULE group should contain the following ODL for each extended criterion used in the search:

```
GROUP = SPECIALIZED_RESULTS
  RESULT_NAME =
  RESULT_VALUE =
END_GROUP = SPECIALIZED_RESULTS
```

At a minimum, this information should be displayed in granule comments. It would be better to provide searched-on parameters in the inventory results summary screen, instead of hiding it in the detailed information.

We could get more use out of this feature if it were not tied to criteria, but could accommodate any kind of special result whether it's searchable or not, and if the value allowed is long enough. Specifying a separate SPECIALIZED_RESULT group with RESULT_NAME and RESULT_VALUE might avoid problems when the values for fields cannot be narrowed to a discrete list of values (and therefore are difficult to allow for searching), and the value length might be large (like a large, verbose comments field).

Specialized Search Applications at the Data Centers

Many data centers have already implemented specialized search functions within their local user interfaces. Those data centers may supply the URL for a data center's Web application to the IMS client. A user desiring additional search capabilities can follow the appropriate link and be transferred to the data center's search application. This is one of several types of URLs to be supported by the V0 IMS Web Gateway.

Recommendations for use of the SPECIALIZED_SEARCH_URL:

Data Centers can supply Specialized Search URLs to the Client team along with the URLs for other DAAC-unique functions. These URLs are typically available from the IMS Welcome Screen. Like data URLs, a SPECIALIZED_SEARCH_URL group can be included within the appropriate DATASET group of a Directory Result or Inventory Result message. In this case, availability of dataset-specific search should be indicated on the results screen (dataset list) of the Web gateway. ODL structure for the specialized search URL:

```
GROUP = SPECIALIZED_SEARCH_URL
  URL =
  URL_COMMENT =
```

END_GROUP = SPECIALIZED_SEARCH_URL

Ideally, a data center's dataset-specific search application will include an order function as well. Each data center's Web application should have a link back to the IMS Web gateway.

No other IMS messages will be affected in this implementation.

Back to Message Summary

Author: Helen Conover, UAH - helen.conover@msfc.nasa.gov

Responsible NASA Official: Yonsook Enloe, NASA/GSFC - yonsook@harp.gsfc.nasa.gov

Last Updated: 24 March 1997