

Semantic Interoperability Scripting and Measurements

Ke-Thia Yao, In-Young Ko, Robert
Neches, Robert MacGregor
University of Southern California
Information Sciences Institute



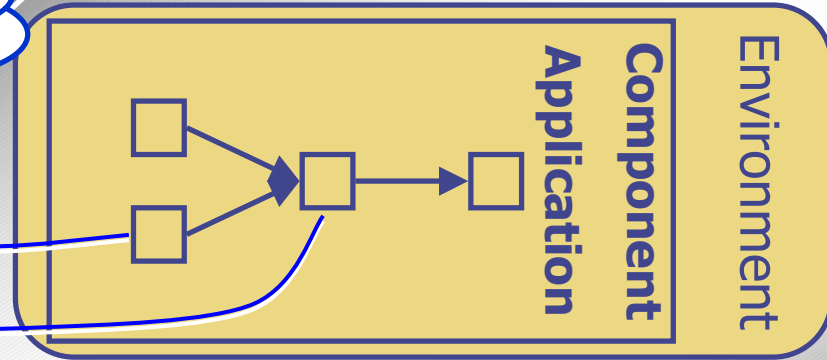
Component-based Software Development

- ◆ Vision of component-based software design is to assemble software applications from a database of existing software components
- ◆ Paradigm explicitly partitions software design task
 - Component developers that build the component database
 - Application developers that assemble the software
 - Administrators that maintain the software
- ◆ Partitioning of tasks increases demand on the exchange of design information among the parties

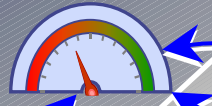


Component Software Development Process

How can I build a correct application with these components?



Application Developer



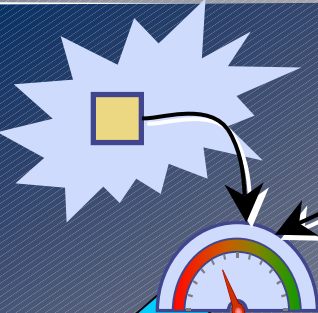
Component Database

- by function, by I/O data

Architectural Styles

- centralized, distributed
- synchronous, asynchronous

Component Framework



Component Developer

How do I make a new component interoperable?

How do I best configure this application for my local environment?

System Architect/
Administrator



Component Software Development Support

- ◆ Software development tools should explicitly support the automated exchange and reasoning of design information
- ◆ Current tools such as Interface Description Languages (IDLs) help to a point
 - Assume exact matching on data types and behaviors
 - ◆ Plug-compatible substitutions: *yes*
 - ◆ Adaptation of closely related components: *no*
 - Capture interface and method specifications
 - ◆ Syntax level composition: *yes*
 - ◆ Qualitative considerations in composition: *no*
 - Examples: implementation effort, performance, semantics



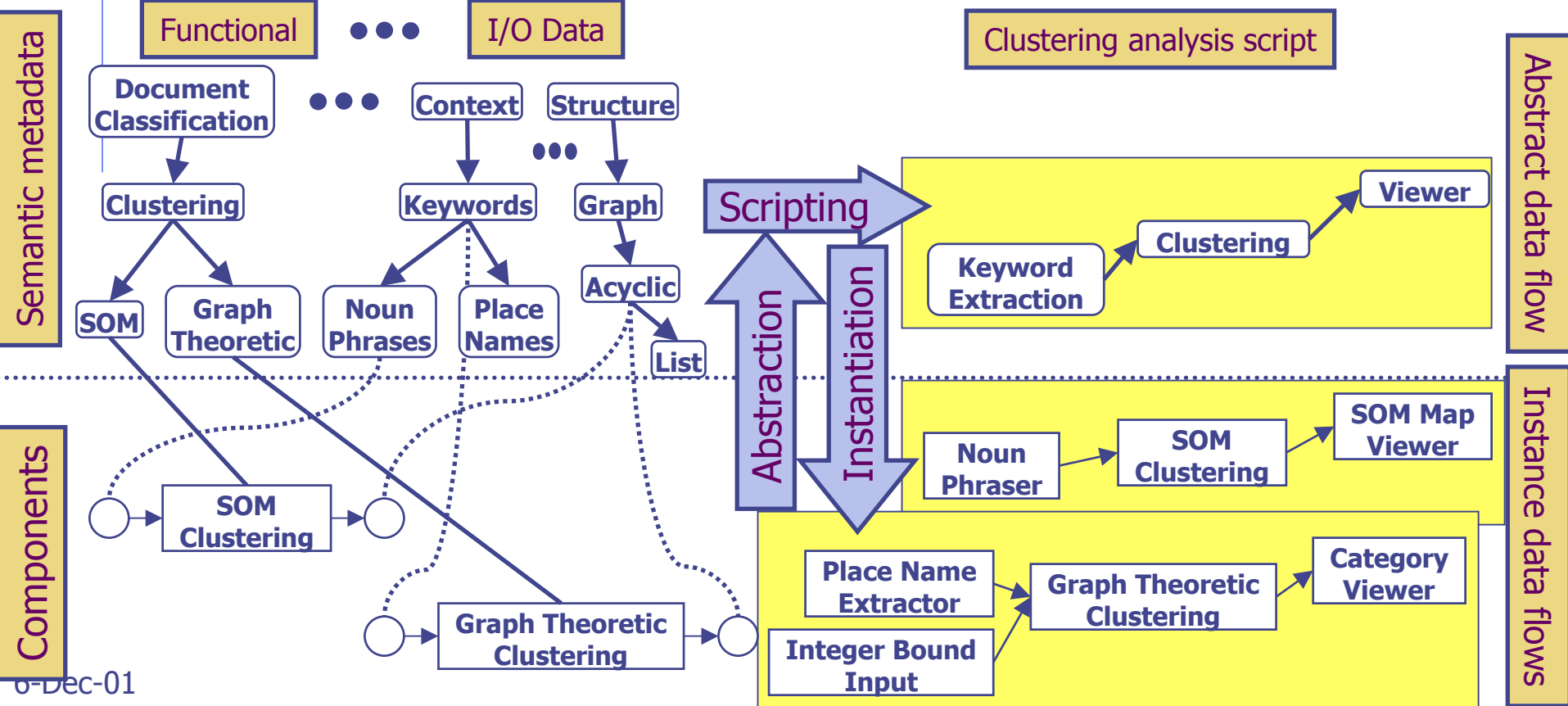
Approach

1. Provide better component/system metadata
 - Help designers express what they have created
 - Help other designers understand what they're working with
2. Provide metadata-level scripting mechanism
 - Help designers assemble software applications
3. Provide software gauges
 - Help application developers make component selections
 - Help component developers create new components and maintain the component database
 - Help system architects/administrators make application adaptations

What Do We Mean By Metadata And Scripts



- ◆ Abstraction: semantic description of software components
- ◆ Scripting: metadata-level description of software application
- ◆ Instantiation: instantiation and execution of system-level scripts



Testbed Application: GeoWorlds

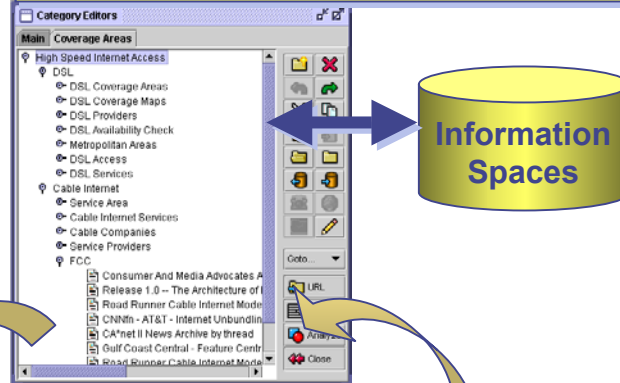
Component Framework



Information Gathering



Information Organization

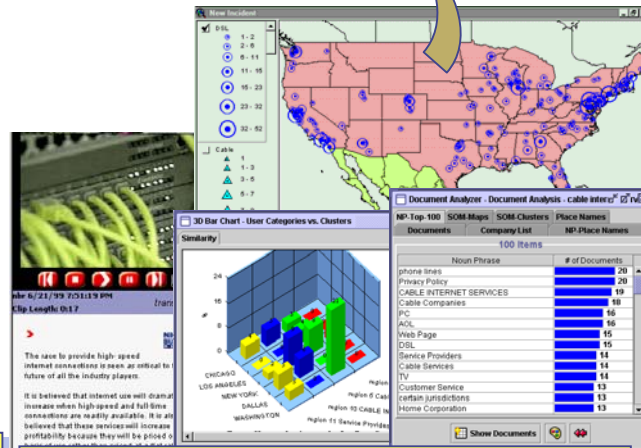
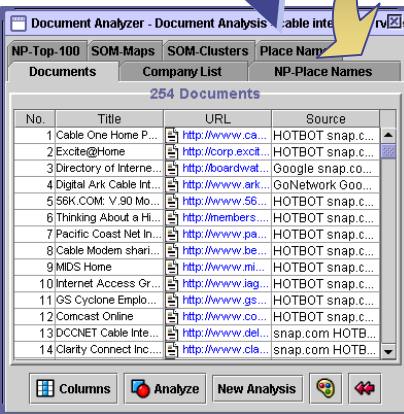


GeoWorlds

- ◆ Geographic Information Systems plus Web processing
- ◆ Retrieve, analyze, visualize and organize documents
- ◆ Component-based system used at US Pacific Command HQ

Useful testbed

- ◆ Actively-used framework for adding, executing components
- ◆ Heavy emphasis on runtime application composition
- ◆ Large number of components, which stress tests
 - Adding new components
 - Building/changing component applications



Document Analysis

Information Visualization

Metadata Level Modeling

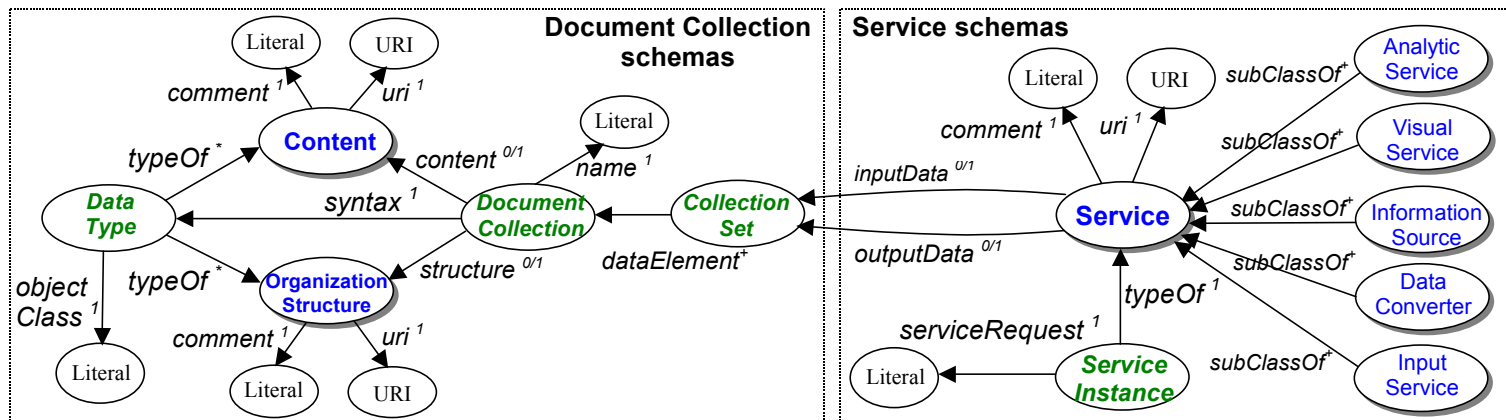


Goals:

- ◆ Extensible metadata that is computational efficient
- ◆ Human understandable and easy to maintain
- ◆ Enables comparison of components wrt:
 - Functionality, I/O data

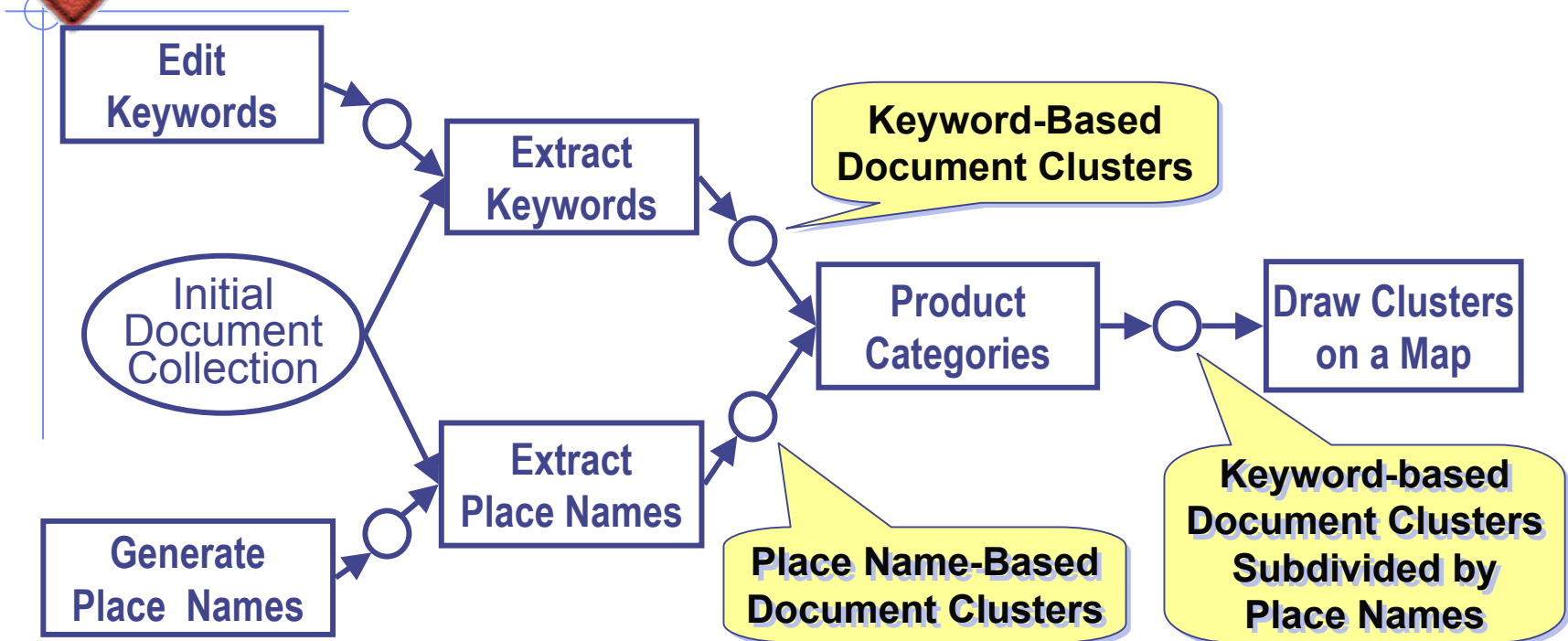
Metadata framework:

- ◆ Characteristics:
 - Lightweight, Multi-form ontology
 - Subsumption-based inferencing
- ◆ Organized by hierarchies
 - Data: document collections e.g., *content* and *structure*
 - Services: classes of operations, e.g., *Analytic*, *visual*, *information source*, *data converter*, *input*





Semantically-based Service Scripting



- ◆ Define software application by defining data-flow among components
- ◆ Ensure generation of semantically valid scripts
- ◆ Automatically insert data converters
- ◆ Instantiate, execute metadata-level scripts



Script Example: China Disaster Analysis

Retrieve a Web document collection about "China disasters"

No.	Title	URL	Source
1	New snows wide...	http://www.tib...	Google EXCITE
2	Disaster Relief fro...	http://www.di...	AllTheWeb EXCI...
3	China Disaster Relief	http://redcross...	AllTheWeb AltaV...
4	EMERGENCY RELI...	http://www.jc...	snap.com HOT...
5	1997 World Disast...	http://www.qu...	GoNetwork AllT...
6	China's one-child p...	http://www.ne...	AllTheWeb EXCI...
7	Suez Lyonnaise d...	http://www.v...	snap.com AllTh...
8	BBC News Asia...	http://news.bb...	HOTBOT snap.c...
9	The Price of 'enga...	http://www.fr...	AllTheWeb EXCI...
10	USAID Press Rele...	http://www.us...	AllTheWeb Goo...
11	Clinton's China Poll...	http://frontpag...	AltaVista AllThe...
12	Unbelievable polls ...	http://www.fr...	EXCITE Google
13	Emergency Aid to t...	http://www.m...	snap.com HOT...
14	CHINA CHARITY F...	http://www.ch...	snap.com HOT...

Classify documents based on the disaster types mentioned

- Document clusters for 'China disaster'
 - QUAKE
 - FIRE
 - FLOODING
 - 1997 World Disaster Reduction
 - USAID Provides Disaster Assi
 - NATIONAL INTERAGENCY COO
 - China Floods Exacerbated By M

Cross-product between place names and the disaster type categories

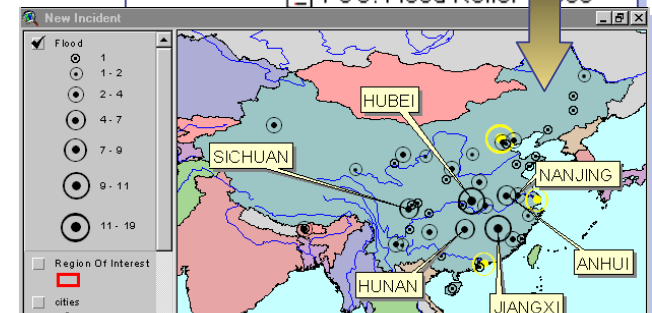
- China disasters
 - FIRE
 - QUAKE
 - FLOOD
 - ANHUI
 - ANSHAN
 - BEIJING
 - CHENGDU
 - CHONGQING
 - DUSHANBE
 - FUJIAN
- Catastrophic Flood Disaster
- FCC: Flood Relief - 1998

- Document clusters for 'China disaster'
 - BEIJING
 - CHENGDU
 - CHONGQING
 - GUANGZHOU
 - HANGZHOU
 - HARBIN
 - HONG KONG
 - JILIN
 - NATIONAL INTERAGENCY COO
 - Catastrophic Flood Disaster in

Get a set of place names from the map and classify the documents based on them



Plot the document clusters on the map to figure out the major flooding areas





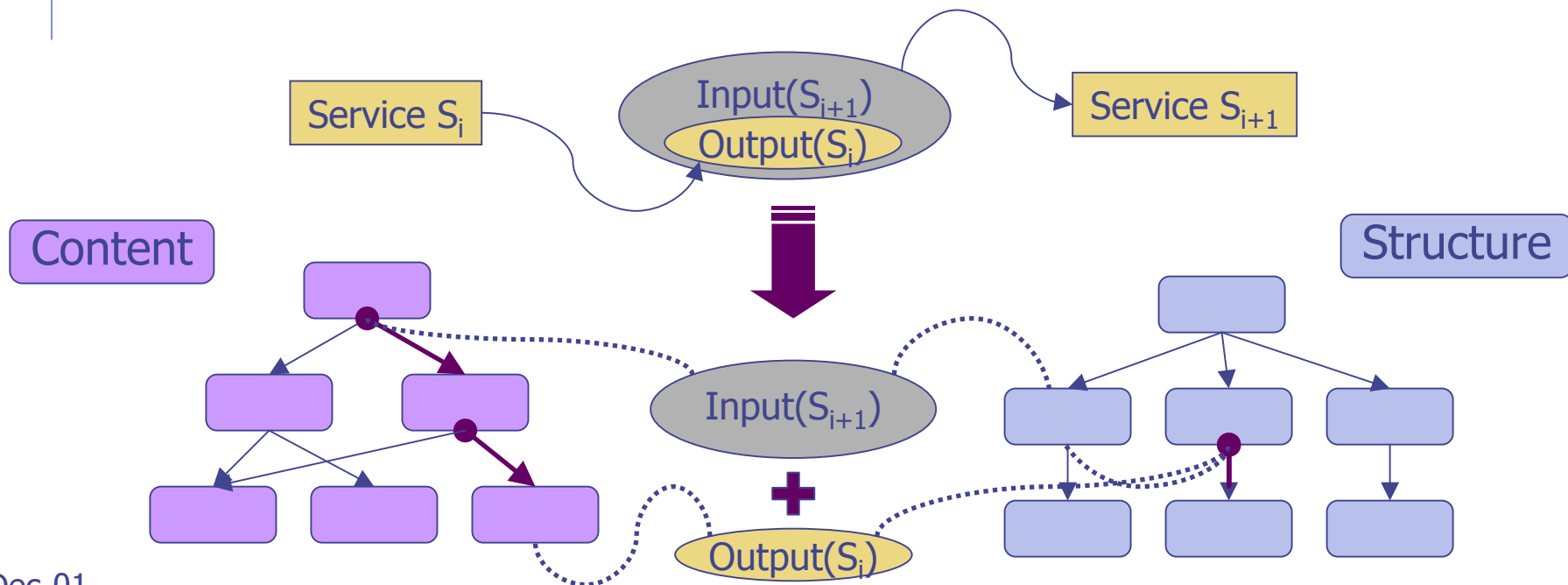
Software Gauges

- ◆ *Interoperability gauges* to find next components in the script generation process
- ◆ *Compatibility gauges* to adapt scripts by finding candidate replacement components
- ◆ *Insertion gauges* to judge
 - Interoperability: the level of effort required to insert new components into the repository
 - Utility: the uniqueness of new components within the repository



Measures Used by Interoperability Gauges: Subsumption and Graph Distance

- ◆ Which services handle the output of the current service?
 - The input of the service must subsume the output of the current service wrt content and structure
- ◆ How well do they match?
 - Prefer more specific matches by using graph distance



Nested Interoperability Gauges

Categorization gauge

- Finds all services that can be connected to the current selected services
- Categorize services based on their functionality

Quality gauge

- Displays degree of interoperability

GeoWorlds example:

- ◆ Suggest *map viewer* to display geographical data
- ◆ If frequency available,
 - Rank *frequency list viewer* over *generic list viewer*, and over *category viewer*
- ◆ Suggest specialized services
 - Suggest *SOM clustering* after *Noun Phraser*

The screenshot displays the Service Composer interface. The top part shows a workflow diagram with components like 'Info Source', 'Analyzer', and 'GW_Categorization'. The bottom part shows a detailed view of a 'PlaceNameBasedClassification' service. This view includes a table of data items with columns for 'Name', 'Content', and 'Structure'. The table lists items like 'Place' and 'Noun' with their respective categories and actions.

Name	Content	Structure
Place		
Noun		

Problem identification gauge

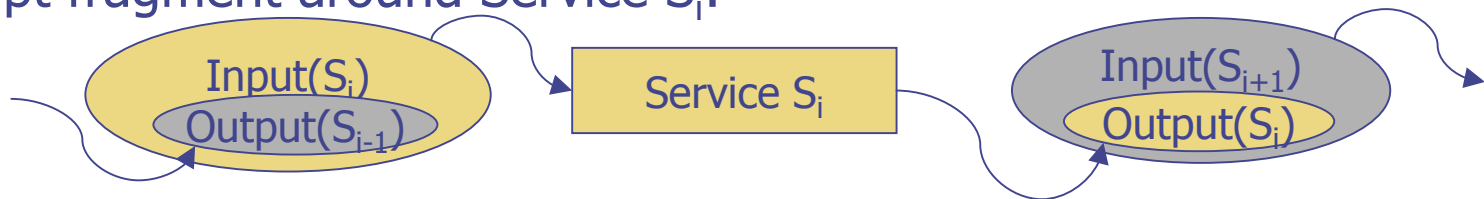
- identifies sources of interoperability problems



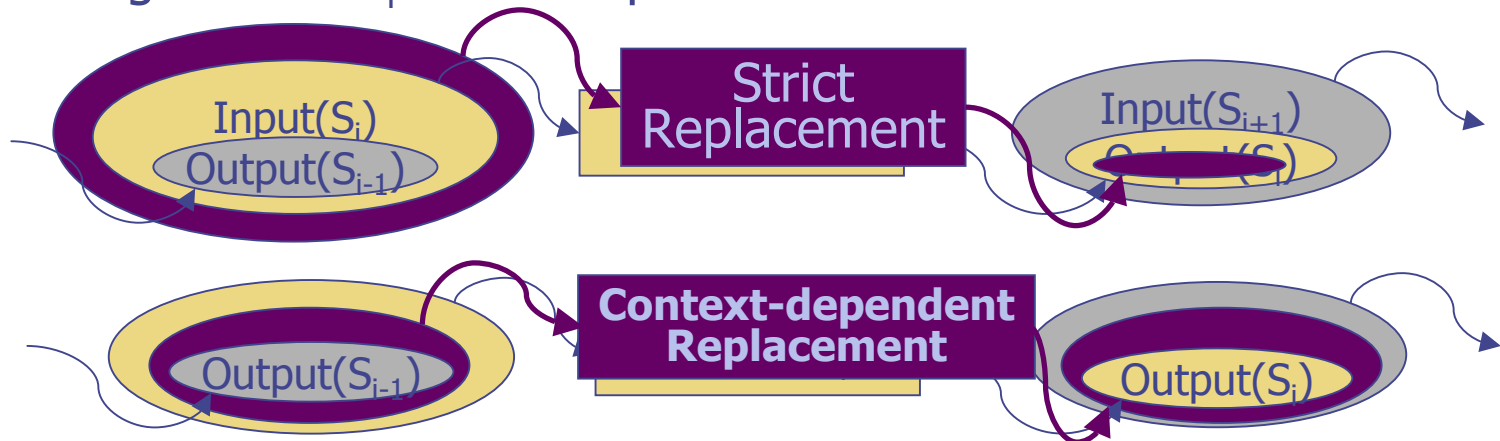
Measures Used by Compatibility Gauges: Context-dependent Semantics, Not Strict Semantics

- Which services can accept the predecessor's output and conform to the successor's input parameter ?
 - Script context permit less stringent constraints on replacement services
 - Automatic insertion of data converter services

Script fragment around Service S_i :



Replacing Service S_i with compatible services :





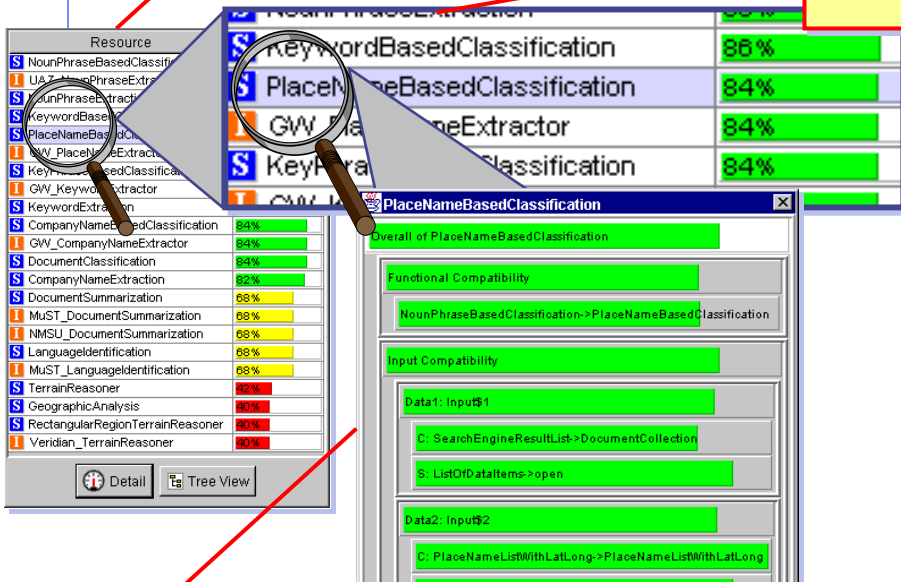
Nested Compatibility Gauges

Ranking gauge

- ◆ Finds all replacement services satisfying the context-dependence semantics
- ◆ Ranks replacement services by distance from the original service

Quality gauge

- Displays degree of compatibility



GeoWorlds example:

- ◆ *Graph-based clustering* requires the output of *Noun Phraser*
 - *Keyword Extractor* is not strictly compatible with *Noun Phraser*
 - It is context-dependent compatible, because *Graph-based clustering* can accept its output

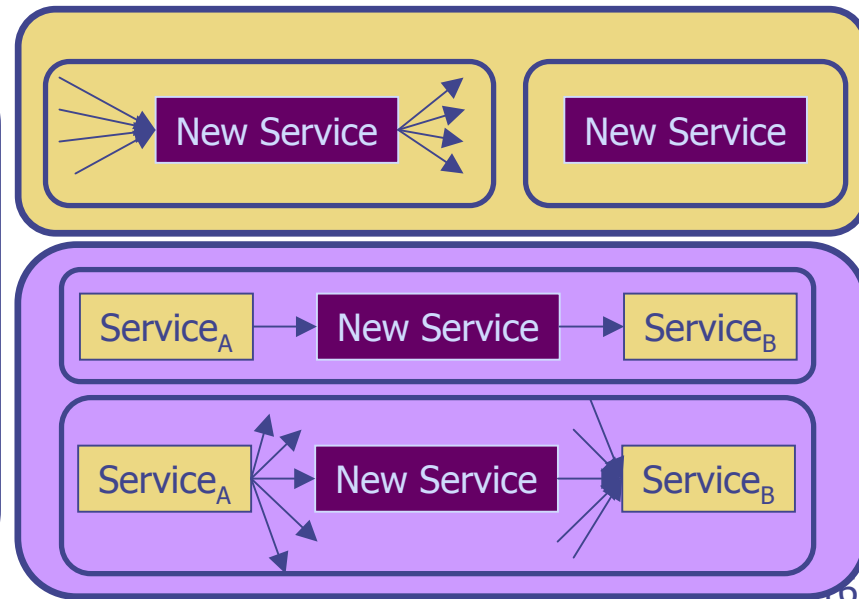
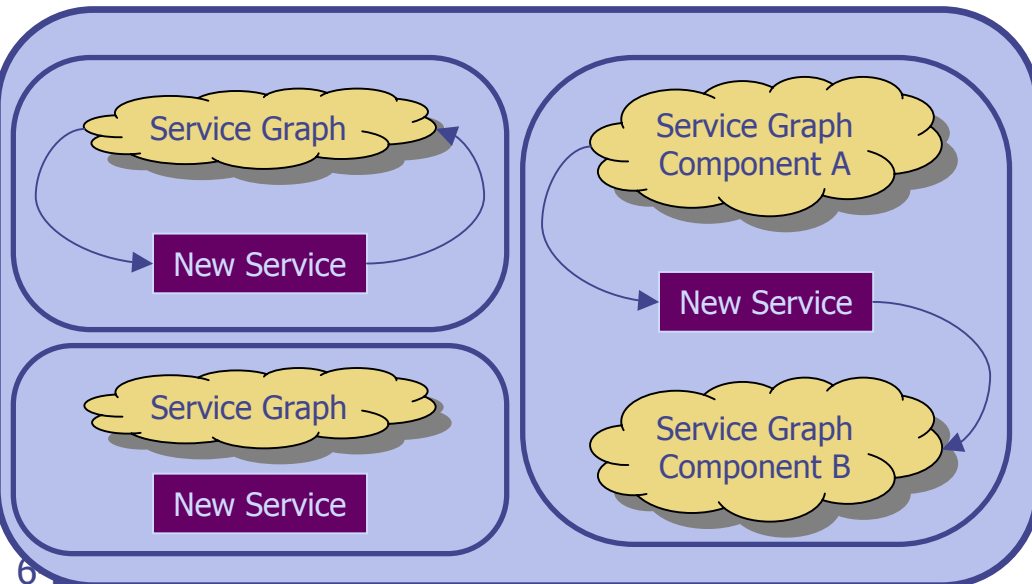
Problem identification gauge

- identifies sources of compatibility problems

Measures Used by Insertion Gauges: Node and Graph Connectivity



- ◆ How interoperable is the new service with the existing services?
 - Service connectivity graph: each node represents a service, arc from node A to B indicate B accepts A's output
 - Measure interoperability using number of connected components and branching factor
- ◆ What does the new service contribute? Unique functionality versus redundancy?
 - Use compatibility measure to compare against existing services

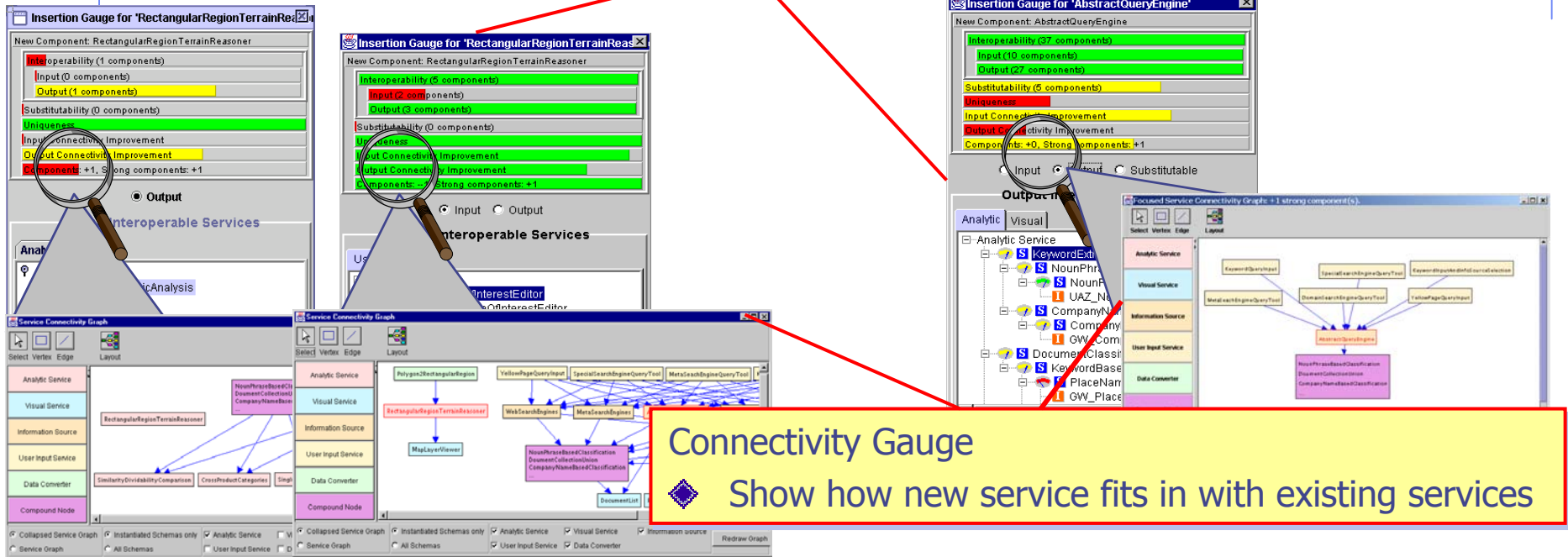




Nested Insertion Gauges

Insertion overview Gauge

- ◆ Compares new services against existing services for interoperability, uniqueness/substitutability



Connectivity Gauge

- ◆ Show how new service fits in with existing services

Veridian's Terrain Reasoner

- Low interoperability, high uniqueness, low graph and node connectivity

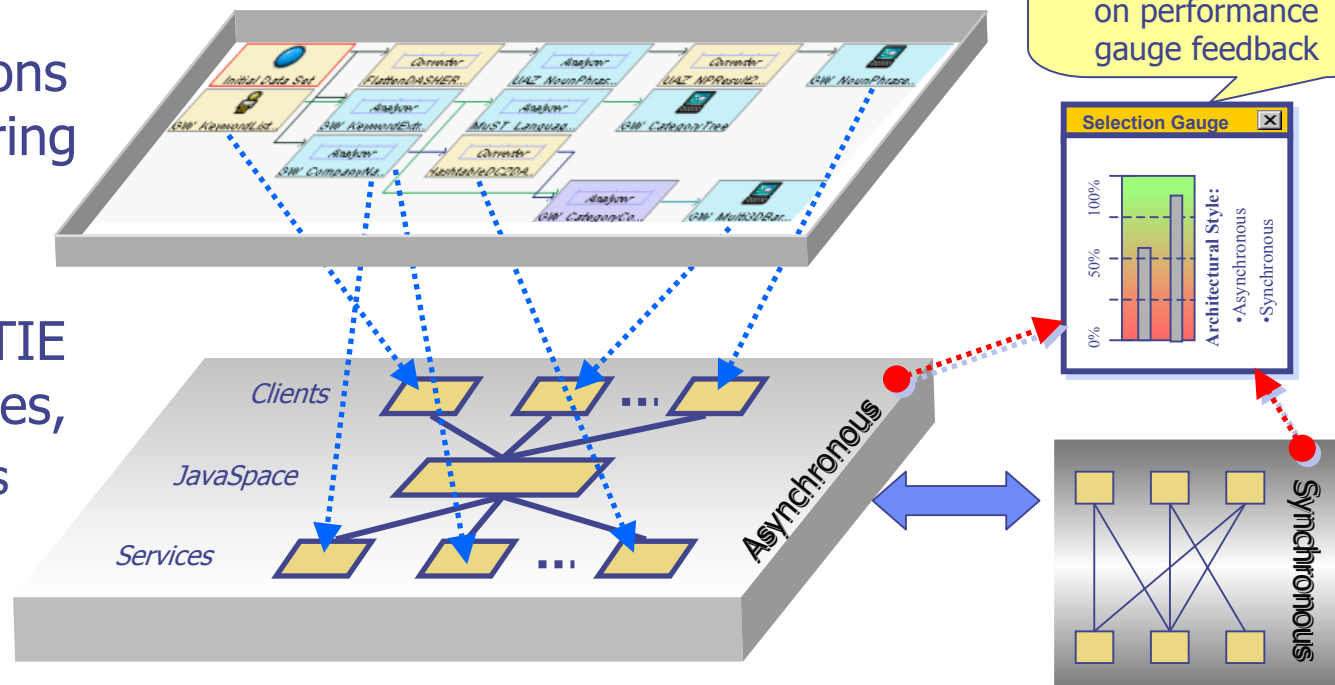
BBN's AQE

- High interoperability, low uniqueness, good graph and node connectivity

Future Work: Architectural-level Gauges



- ◆ Improve performance by applying architectural level transformations
 - ◆ Transformations must be *semantically invariant* wrt to script
 - Physical component redeployment
 - Connector change
 - ◆ Transformations can occur during runtime
 - ◆ Guided by IntelliGauge TIE runtime gauges,
 - bottlenecks and faults





Example: GeoTopics Application Built using GeoWorlds Framework

GeoTopics, Thursday, October 25, 2001 11:00 AM PDT - Netscape

File Edit View Go Communicator Help

Bookmarks Netsite: <http://www.isi.edu/geoworlds/geotopics/bydate/date200110251100/> What's Related

GeoTopics
The hot topics around the world brought to you by GeoWorlds
A RESEARCH PROJECT OF
[University of Southern California](#)
School of Engineering
Information Sciences Institute
Distributed Scalable Systems Division

USC
UNIVERSITY OF SOUTHERN CALIFORNIA

[Home](#) | [About Us](#) | [Contact Us](#) | [Credits](#) | [How it works](#) | [News media included](#) | [First time users](#) | [Help](#)

Previous Thursday, October 25, 2001 11:00 AM PDT Next

Top 20 Topics

1	↑ [2]	Anthrax Cases (95)	
2	↓ [1]	Terror Attacks (88)	
3	NEW	Attacks on Afghanistan... (68)	
4	↑ [3]	Taliban Regime (62)	
5	↓ [4]	Osama bin Laden (60)	
6	↓ [3]	George W. Bush (49)	
7	↑ [13]	War on Terrorism (41)	
8	NEW	Northern Alliance (30)	
9	NEW	World Trade Center (33)	
10	↓ [6]	Biochemical Attacks (32)	
11	↓ [8]	White House (28)	
12	↓ [17]	Economic Recovery (24)	
13	↓ [12]	FBI (21)	
14	↓ [15]	Public Health (20)	
15	↓ [16]	Bush administration (25)	
16	NEW	Windows XP (16)	
17	NEW	United Nations (19)	
18	NEW	Secretary of State Co... (20)	
19	↓ [10]	Senate Majority Leade... (18)	
20	NEW	Former President Ch... (17)	

Top 20 Places

1	↑ [1]	UNITED STATES (171)	
2	↑ [2]	NEW YORK (95)	
3	↑ [3]	AFGHANISTAN (78)	
4	↑ [4]	PAKISTAN (45)	
5	↑ [11]	KABUL (33)	
6	↓ [5]	WASHINGTON D.C. (34)	
7	↑ [9]	LOS ANGELES (30)	
8	↑ [12]	SAN FRANCISCO (24)	
9	↑ [10]	CHICAGO (76)	
10	↑ [4]	IRAQ (18)	
11	↑ [6]	ISRAEL (19)	
12	↑ [8]	LONDON (18)	
13	↑ [17]	GERMANY (15)	
14	NEW	IRAN (15)	
15	↑ [15]	UNITED KINGDOM (15)	
16	↓ [7]	MIAMI (21)	
17	NEW	FRANCE (12)	
18	NEW	ITALY (11)	
19	↑ [19]	SAUDI ARABIA (11)	
20	↑ [20]	WEST BANK (12)	

Legend

■ Same rank ↑ Moving up [XX] Number of articles ■ Articles on Thursday, October 25, 2001 11:00 AM PDT
 ■ New today ↓ Moving down [XX] Rank yesterday ■ Articles on Wednesday, October 24, 2001 7:00 PM PDT

Document: Done

- ◆ Monitors daily news articles from leading English language papers
- ◆ Identifies hot topics, hot places, and how the two relate
- ◆ Useful testbed
 - GeoTopics executes daily provide probing opportunities
 - Components executed remotely on distributed servers
 - ◆ Monitor server performance
 - Need to access remote information sources
 - ◆ Monitor network congestion/ interruptions (throws Java exceptions)
- ◆ www.isi.edu/geoworlds/geotopics



Summary

We have developed

- ◆ Metadata-level description of software components
 - Provides behavior and input/output description at more abstract level
 - Enables reasoning about their interoperability
- ◆ Metadata-level and system-level scripting mechanism
 - Ensure generation of semantically valid applications
 - Provides script instantiation and execution
- ◆ Software gauges
 - Interoperability gauge: to help building component applications
 - Compatibility gauge: to help making changes to applications
 - Insertion gauge: to help adding new components