

SFWR ENG 3A04: Software Design II

Dr. Ridha Khedri

Department of Computing and Software, McMaster University
Canada L8S 4L7, Hamilton, Ontario

Term 1, 2008–2009

Acknowledgments: Material based on *Software Architecture Design* by Tao et al. (Chapter 6)

Outline of Part I

- 1 Batch Sequential
- 2 Pipe and Filter Architecture
- 3 Process-Control Architecture
- 4 Data flow model to class model

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Outline

Part I: Review of
Previous Lecture

Part II: Today's
Lecture

Outline of Part II

- 5 Overview
- 6 Repository Architecture Style
- 7 Blackboard Architecture Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Outline

Part I: Review of
Previous Lecture

**Part II: Today's
Lecture**

Part I

Review of Previous Lecture

Part II

Today's Lecture

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store
- The data store is shared by all related software components

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store
- The data store is shared by all related software components
- The software system is decomposed into two major partitions

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The **data centered software architecture**

- It is characterized by a centralized data store
- The data store is **shared** by all related software components
- The software system is decomposed into two major partitions
 - **data store**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The **data centered software architecture**

- It is characterized by a centralized data store
- The data store is **shared** by all related software components
- The software system is decomposed into two major partitions
 - data store
 - **independent software component agents**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store
- The data store is shared by all related software components
- The software system is decomposed into two major partitions
 - data store
 - independent software component agents
- The connections between the data modules and software components are implemented by

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store
- The data store is shared by all related software components
- The software system is decomposed into two major partitions
 - data store
 - independent software component agents
- The connections between the data modules and software components are implemented by
 - explicit method invocation

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

The data centered software architecture

- It is characterized by a centralized data store
- The data store is shared by all related software components
- The software system is decomposed into two major partitions
 - data store
 - independent software component agents
- The connections between the data modules and software components are implemented by
 - explicit method invocation
 - implicit method invocation based on the repository category

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

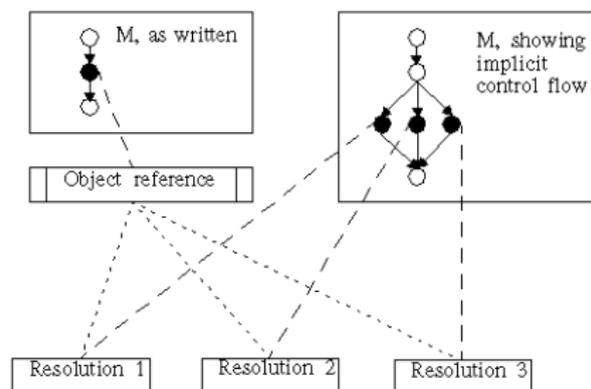


Figure: Implicit control flow

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**
- The shared data module provides all mechanisms for software components to access it

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**
- The shared data module provides all mechanisms for software components to access it
 - **insertion**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**
- The shared data module provides all mechanisms for software components to access it
 - insertion
 - **deletion**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**
- The shared data module provides all mechanisms for software components to access it
 - insertion
 - deletion
 - **update**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

In **pure** data centered software architecture

- The software components do not communicate with each other directly
- All the communications are conducted via **data store**
- The shared data module provides all mechanisms for software components to access it
 - insertion
 - deletion
 - update
 - **retrieval**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow control)
 - Client may access a repository
 - interactively
 - by a batch transaction request

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively
 - by a batch transaction request
 - Repository is widely used in

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively
 - by a batch transaction request
 - Repository is widely used in
 - database management system

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively
 - by a batch transaction request
 - Repository is widely used in
 - database management system
 - library information system

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively
 - by a batch transaction request
 - Repository is widely used in
 - database management system
 - library information system
 - Computer Aided Software Engineering (CASE)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(differentiated by the flow control strategy)

- Repository
 - The data store is passive
 - Clients of the data store are active (software component are taking control of flow logic)
 - Client may access a repository
 - interactively
 - by a batch transaction request
 - Repository is widely used in
 - database management system
 - library information system
 - Computer Aided Software Engineering (CASE)
- Blackboard

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

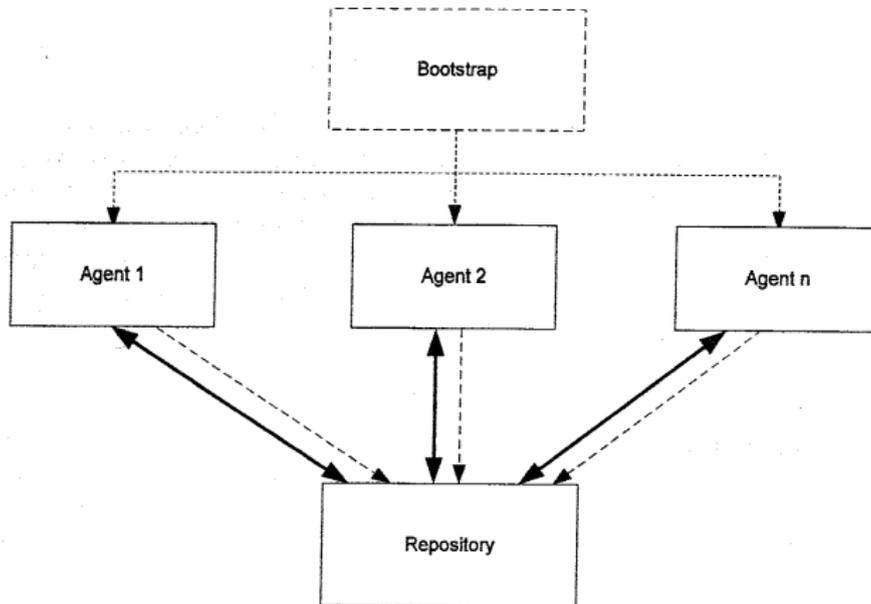


Figure: Repository architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(Continued)

- Blackboard

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(Continued)

- Blackboard
 - The data store in blackboard architecture is active

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(Continued)

- Blackboard
 - The data store in blackboard architecture is active
 - Its clients are passive that the flow logic is determined by the current data status in the data store

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

There are two categories of data centered architecture
(Continued)

- Blackboard
 - The data store in blackboard architecture is active
 - Its clients are passive that the flow logic is determined by the current data status in the data store
 - The clients of a blackboard are called knowledge sources

There are two categories of data centered architecture
(Continued)

- Blackboard
 - The data store in blackboard architecture is active
 - Its clients are passive that the flow logic is determined by the current data status in the data store
 - The clients of a blackboard are called knowledge sources
 - A new data change may trigger events to its listeners (subscribers)

There are two categories of data centered architecture
(Continued)

- Blackboard (Continued)

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(Continued)

- Blackboard (Continued)
 - A new data change may drive the flow logic continually until a goal is reached

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture Overview

There are two categories of data centered architecture
(Continued)

- Blackboard (Continued)
 - A new data change may drive the flow logic continually until a goal is reached
 - Most software applications designed in the blackboard architecture are knowledge based systems

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

There are two categories of data centered architecture
(Continued)

- Blackboard (Continued)
 - A new data change may drive the flow logic continually until a goal is reached
 - Most software applications designed in the blackboard architecture are knowledge based systems
 - voice and image recognition system

There are two categories of data centered architecture
(Continued)

- Blackboard (Continued)
 - A new data change may drive the flow logic continually until a goal is reached
 - Most software applications designed in the blackboard architecture are knowledge based systems
 - voice and image recognition system
 - security system

Data Centered Software Architecture Overview

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

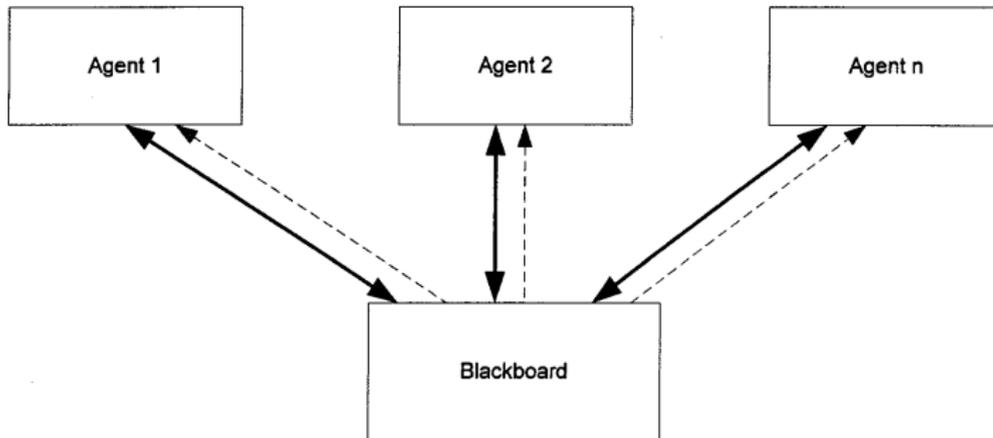


Figure: Blackboard architecture

Data Centered Software Architecture

Repository Architecture Style

- It supports user interaction for data processing instead of batch sequential transaction processing

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- It supports user interaction for data processing instead of batch sequential transaction processing
- The software component agents of the data store control the computation and flow logic of the system

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- It supports user interaction for data processing instead of batch sequential transaction processing
- The software component agents of the data store control the computation and flow logic of the system
- Clients can get data from data store and put data in the store

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- It supports user interaction for data processing instead of batch sequential transaction processing
- The software component agents of the data store control the computation and flow logic of the system
- Clients can **get** data from data store and **put** data in the store
- Different clients may have different interfaces and different privileges

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

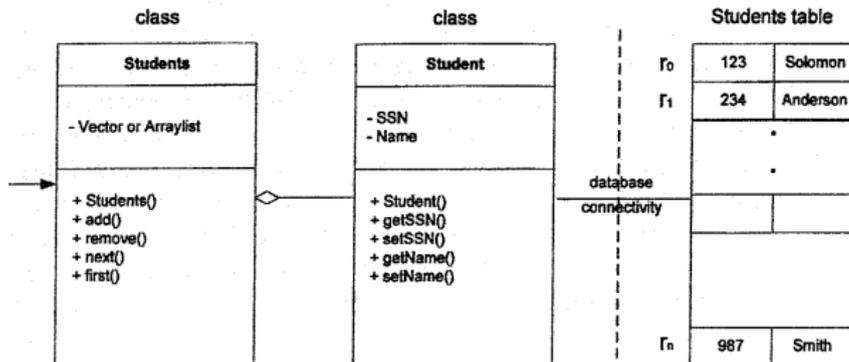


Figure: Student management system (simplified class diagram)

Data Centered Software Architecture

Repository Architecture Style

The Student class is a data bean class which is backed up by a corresponding table in a database

```
public class Student implements Serializable {
    String SSN;
    String Name;

    public Student() {
        SSN = '' '';
        Name = '' '';
    }

    public Student(String ssn, String name) {
        SSN = ssn;
        Name = name;
    }

    public void setSSN(String ssn) { SSN = ssn; }
    public String getSSN() { return SSN; }
    public void setName(String name) { Name=name;}
    public String getName() { return Name; }
}
```

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

The connection between the data bean class and an Oracle database

```
...
...
try {
    Class.forName('sun.jdbc.odbc.JdbcOdbcDriver');
    Connection connection =
        DriverManager.getConnection(
            "jdbc:odbc:students");
}

catch(Exception e){ ... }
```

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

JDBC SQL statement to select all records from the students table and place them in the ResultSet

```
ArrayList studentList = new ArrayList() ;

Statement statement = connection.createStatement () ;
ResultSet results =
    statement.executeQuery("SELECT * FROM students''") ;
Student student = new Student() ;

while (results . next ()) {
    student.setSsn(results.getSsn(1)) ;
    student.setName(results.getName(2)) ;
    studentList.add(student) ;
}
```

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

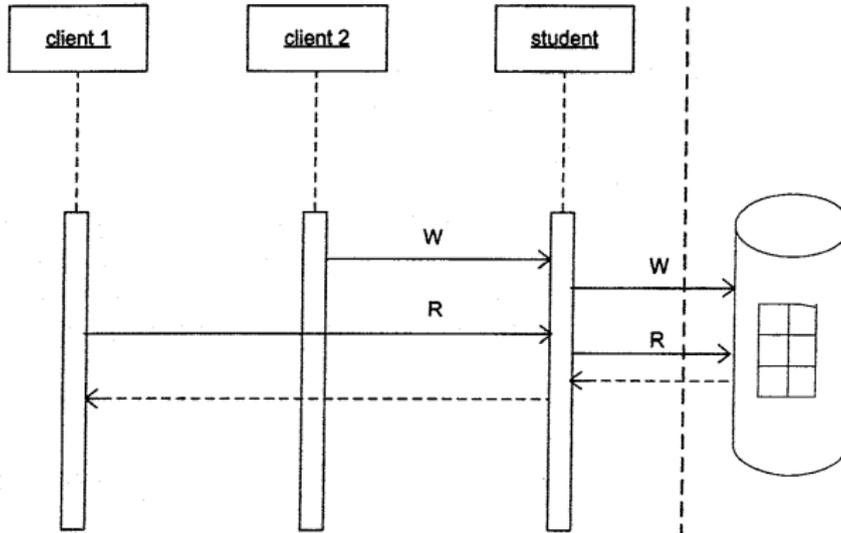


Figure: Dynamic view of this repository architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The relational database management system is a typical design domain for the repository architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The **relational database management system** is a typical design domain for the repository architecture
- The data store of the repository maintains all types of data including schema (meta-data), data tables, and index files for data tables

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The **relational database management system** is a typical design domain for the repository architecture
- The data store of the repository maintains all types of data including schema (meta-data), data tables, and index files for data tables
- **Many tools available to develop application on the database stored in database management system**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The **relational database management system** is a typical design domain for the repository architecture
- The data store of the repository maintains all types of data including schema (meta-data), data tables, and index files for data tables
- Many tools available to develop application on the database stored in database management system
 - **Design tools, development tools, maintenance tools, and documentation tools**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The **relational database management system** is a typical design domain for the repository architecture
- The data store of the repository maintains all types of data including schema (meta-data), data tables, and index files for data tables
- Many tools available to develop application on the database stored in database management system
 - Design tools, development tools, maintenance tools, and documentation tools
 - **Oracle Designer, Oracle Developer, Oracle data warehouse, etc.**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- The **relational database management system** is a typical design domain for the repository architecture
- The data store of the repository maintains all types of data including schema (meta-data), data tables, and index files for data tables
- Many tools available to develop application on the database stored in database management system
 - Design tools, development tools, maintenance tools, and documentation tools
 - Oracle Designer, Oracle Developer, Oracle data warehouse, etc.
 - **Computer Aided Software Engineering (CASE) system** is another design domain for the rep. soft. arch.

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

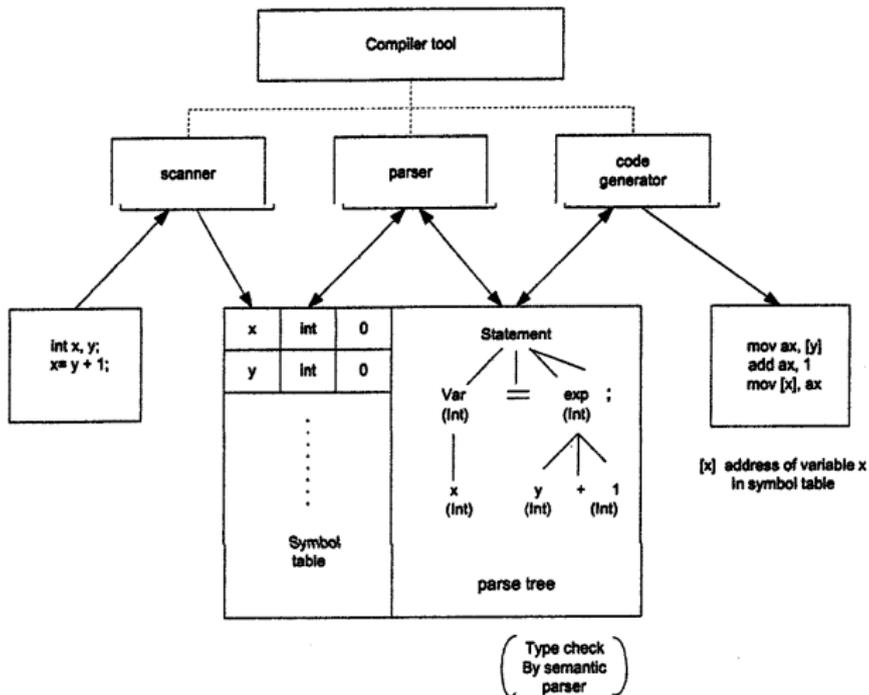
Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style



SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Applicable domain of repository

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Applicable domain of repository
 - Large complex information system where many software component clients need to access it in different aspects

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Applicable domain of repository
 - Large complex information system where many software component clients need to access it in different aspects
- Information system where data transactions drive the control flow of computation

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Applicable domain of repository**
 - Large complex information system where many software component clients need to access it in different aspects
 - Information system where data transactions drive the control flow of computation
- **Benefits**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Applicable domain of repository
 - Large complex information system where many software component clients need to access it in different aspects
 - Information system where data transactions drive the control flow of computation
- Benefits
 - Data integrity: easy to backup and restore

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Applicable domain of repository**
 - Large complex information system where many software component clients need to access it in different aspects
 - Information system where data transactions drive the control flow of computation
- **Benefits**
 - Data integrity: easy to backup and restore
 - **System scalability and Reusability of agents: easy to add new software component**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Applicable domain of repository**
 - Large complex information system where many software component clients need to access it in different aspects
 - Information system where data transactions drive the control flow of computation
- **Benefits**
 - Data integrity: easy to backup and restore
 - System scalability and Reusability of agents: easy to add new software component
 - **Reduce the overhead of transient data between software components**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Limitations

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**
 - Data store reliability & availability are important issues

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- Limitations
 - Data store reliability & availability are important issues
 - Centralized repository is vulnerable to failure compared to distributed repository with data replication

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**

- Data store reliability & availability are important issues
- Centralized repository is vulnerable to failure compared to distributed repository with data replication
- **High dependency between data structure of data store and its agents**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**

- Data store reliability & availability are important issues
- Centralized repository is vulnerable to failure compared to distributed repository with data replication
- High dependency between data structure of data store and its agents
- **The changes of data structure have significant impacts on its agents**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**

- Data store reliability & availability are important issues
- Centralized repository is vulnerable to failure compared to distributed repository with data replication
- High dependency between data structure of data store and its agents
- The changes of data structure have significant impacts on its agents
- **Overhead cost of moving data on network if data is distributed**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**

- Data store reliability & availability are important issues
- Centralized repository is vulnerable to failure compared to distributed repository with data replication
- High dependency between data structure of data store and its agents
- The changes of data structure have significant impacts on its agents
- Overhead cost of moving data on network if data is distributed

- **Related Architecture**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

**Repository
Architecture Style**

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

- **Limitations**

- Data store reliability & availability are important issues
- Centralized repository is vulnerable to failure compared to distributed repository with data replication
- High dependency between data structure of data store and its agents
- The changes of data structure have significant impacts on its agents
- Overhead cost of moving data on network if data is distributed

- **Related Architecture**

- Layered architecture, multi-tier architecture, Model-view-controller (MVC) architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The word blackboard comes from classroom teaching and learning

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The word blackboard comes from classroom teaching and learning
- Teacher and students can share data in solving a hard open problem on classroom blackboard

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The word blackboard comes from classroom teaching and learning
- Teacher and students can share data in solving a hard open problem on classroom blackboard
- Each student and each teacher plays a role of agent to make their contributions to the problem solving

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The word blackboard comes from classroom teaching and learning
- Teacher and students can share data in solving a hard open problem on classroom blackboard
- Each student and each teacher plays a role of agent to make their contributions to the problem solving
- They all can work in parallel and very independently

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The word blackboard comes from classroom teaching and learning
- Teacher and students can share data in solving a hard open problem on classroom blackboard
- Each student and each teacher plays a role of agent to make their contributions to the problem solving
- They all can work in parallel and very independently
- The idea of blackboard architecture is very similar to the classroom blackboard

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- It is a data directed and partial data driven architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- It is a data directed and partial data driven architecture
- The whole system is decomposed into two major partitions

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- It is a data directed and partial data driven architecture
- The whole system is decomposed into two major partitions
 - **blackboard sub-system used to store data (hypotheses and facts)**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- It is a data directed and partial data driven architecture
- The whole system is decomposed into two major partitions
 - **blackboard** sub-system used to store data (hypotheses and facts)
 - **knowledge source** sub-system (domain-specific knowledge is stored)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- It is a data directed and partial data driven architecture
- The whole system is decomposed into two major partitions
 - **blackboard** sub-system used to store data (hypotheses and facts)
 - **knowledge source** sub-system (domain-specific knowledge is stored)
- There may be a third sub-system called **controller** (used to initiate the blackboard and knowledge sources, takes a bootstrap role and overall supervision control)

Data Centered Software Architecture

Blackboard Architecture Style

- The connections between the two partitions are implicit invocation from the blackboard to a specific knowledge source in the blackboard

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The connections between the two partitions are implicit invocation from the blackboard to a specific knowledge source in the blackboard
- The data change in the blackboard triggers a matched knowledge source to continue its reasoning

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- The connections between the two partitions are implicit invocation from the blackboard to a specific knowledge source in the blackboard
- The data change in the blackboard triggers a matched knowledge source to continue its reasoning
- The data change may be caused by new deduced conclusion or hypotheses results by some knowledge sources (called **publish/subscribe mode**)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

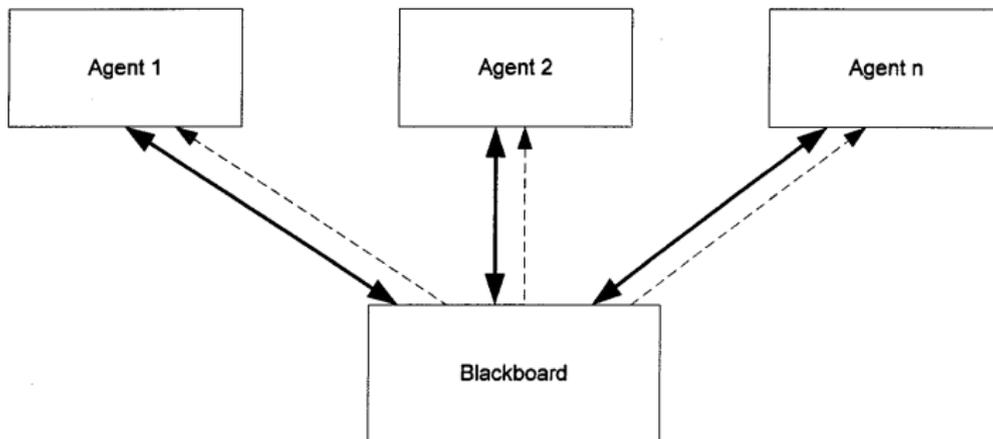


Figure: Blackboard architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Many domain specific knowledge sources collaborate together to solve a complex problem such authentication in information security

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Many domain specific knowledge sources collaborate together to solve a complex problem such authentication in information security
- Each knowledge source is relatively independent from other knowledge source

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Many domain specific knowledge sources collaborate together to solve a complex problem such authentication in information security
- Each knowledge source is relatively independent from other knowledge source
- No need to interact with each other

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Many domain specific knowledge sources collaborate together to solve a complex problem such authentication in information security
- Each knowledge source is relatively independent from other knowledge source
- No need to interact with each other
- They only need to interact and respond to the blackboard sub-system

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Many domain specific knowledge sources collaborate together to solve a complex problem such authentication in information security
- Each knowledge source is relatively independent from other knowledge source
- No need to interact with each other
- They only need to interact and respond to the blackboard sub-system
- Each source only works on a specific aspect of the problem, and contributes a partial solution to the ultimate solution

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

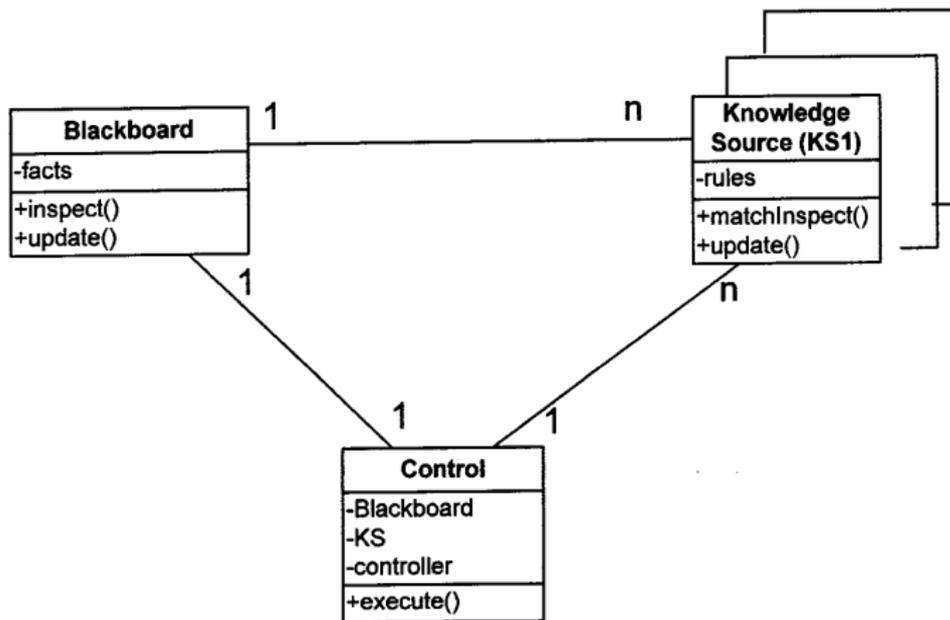


Figure: UML class diagram of rule-based blackboard software architecture

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Repository Architecture Style

Knowledge source (Set of rules)

RI: IF animal gives milk THEN animal is mammal

R2: IF animal eats meat THEN animal is carnivore

R3: IF animal is mammal AND animal is carnivore AND animal has tawny color AND animal has black stripes THEN animal is tiger

Blackboard (The set of facts)

F I: animal eats meat

F2: animal gives milk

F3: animal has black strips

F3: animal has tawny color

- The goal is to recognize an animal (Using forward reasoning or backward reasoning)

Data Centered Software Architecture

Blackboard Architecture Style

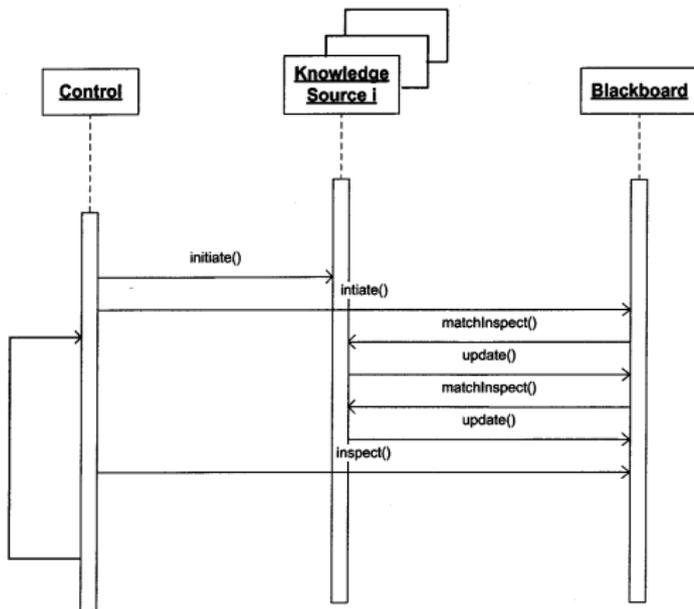


Figure: Sequence diagram of blackboard architecture

Data Centered Software Architecture

Blackboard Architecture Style

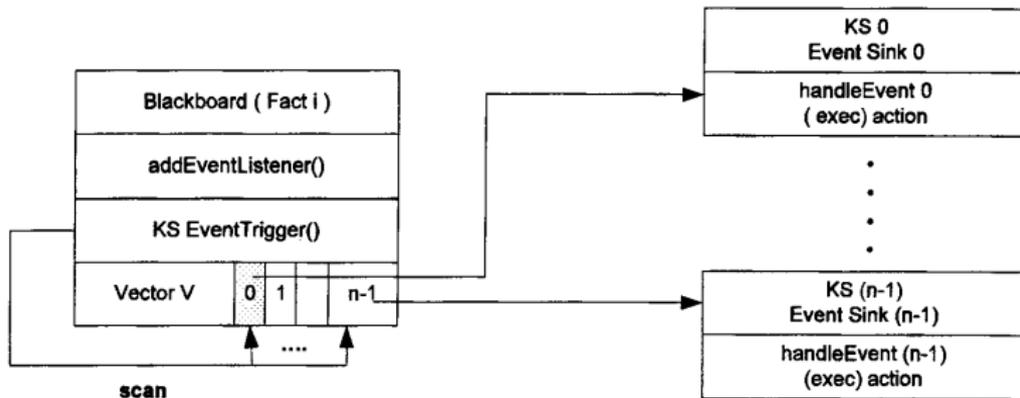


Figure: Publish/subscribe relationship between blackboard and knowledge sources

Data Centered Software Architecture

Blackboard Architecture Style

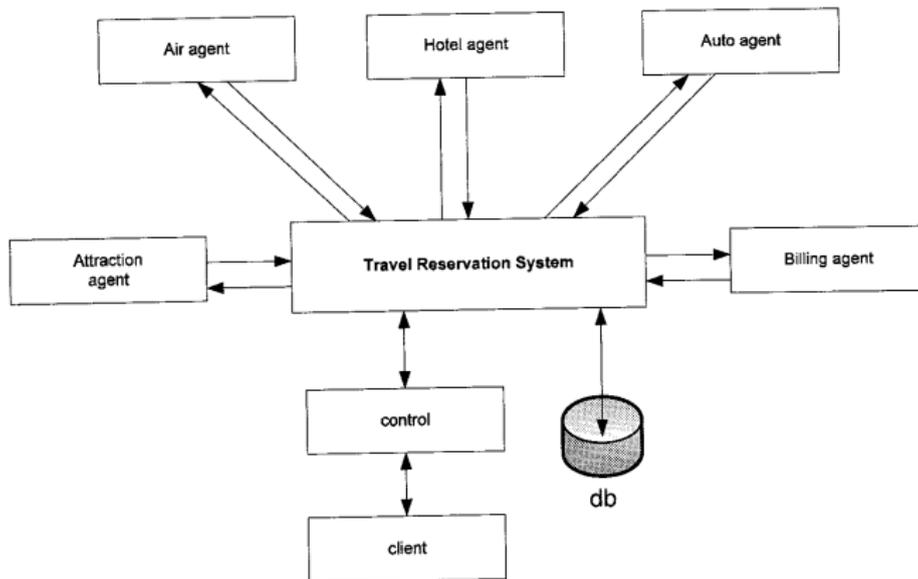


Figure: Blackboard architecture for a travel consulting system

Data Centered Software Architecture

Blackboard Architecture Style

- Applicable domain

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

**Blackboard
Architecture Style**

Data Centered Software Architecture

Blackboard Architecture Style

- Applicable domain
 - Suitable for solving immature and complex problems where non deterministic solutions exist

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
- **The problem spans multiple disciplines**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
 - The problem spans multiple disciplines
 - **Optimal, partial, or approximate solution is acceptable to the problems**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
 - The problem spans multiple disciplines
 - Optimal, partial, or approximate solution is acceptable to the problems
 - **Exhausted searching is impossible and impractical since it may take forever**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
 - The problem spans multiple disciplines
 - Optimal, partial, or approximate solution is acceptable to the problems
 - Exhausted searching is impossible and impractical since it may take forever
- **Benefits**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
 - The problem spans multiple disciplines
 - Optimal, partial, or approximate solution is acceptable to the problems
 - Exhausted searching is impossible and impractical since it may take forever
- **Benefits**
 - **Scalability: easy to add new knowledge source or update existing knowledge source**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Applicable domain**
 - Suitable for solving immature and complex problems where non deterministic solutions exist
 - The problem spans multiple disciplines
 - Optimal, partial, or approximate solution is acceptable to the problems
 - Exhausted searching is impossible and impractical since it may take forever
- **Benefits**
 - Scalability: easy to add new knowledge source or update existing knowledge source
 - **Concurrency: all knowledge sources can work in parallel since they are very independent of each other**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- Benefits (Continued)

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

**Blackboard
Architecture Style**

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - **Reusability of knowledge source agents**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**
 - **Tight dependency between the blackboard and knowledge source**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**
 - Tight dependency between the blackboard and knowledge source
 - Difficult to make a decision when to terminate reasoning

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**
 - Tight dependency between the blackboard and knowledge source
 - Difficult to make a decision when to terminate reasoning
 - **Synchronization of multiple agents is an issue**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**
 - Tight dependency between the blackboard and knowledge source
 - Difficult to make a decision when to terminate reasoning
 - Synchronization of multiple agents is an issue
 - **Debugging and testing of the system is a challenge (no clear execution path)**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)

- Supports experimentation for hypotheses
- Reusability of knowledge source agents

- **Limitations**

- Tight dependency between the blackboard and knowledge source
- Difficult to make a decision when to terminate reasoning
- Synchronization of multiple agents is an issue
- Debugging and testing of the system is a challenge (no clear execution path)

- **Related architecture**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

Data Centered Software Architecture

Blackboard Architecture Style

- **Benefits** (Continued)
 - Supports experimentation for hypotheses
 - Reusability of knowledge source agents
- **Limitations**
 - Tight dependency between the blackboard and knowledge source
 - Difficult to make a decision when to terminate reasoning
 - Synchronization of multiple agents is an issue
 - Debugging and testing of the system is a challenge (no clear execution path)
- **Related architecture**
 - **Implicit invocation architecture such as event-based**

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style

SFWR ENG 3A04:
Software Design II

Dr. R. Khedri

Overview

Repository
Architecture Style

Blackboard
Architecture Style