Data Service
A DAS for services

Luciano Resende

Open Source SOA: SCA, SDO & DAS
http://incubator.apache.org/Tuscany

APACHE
TUSCANY

What is a Data Service

➢ **Data Service is an extension to the SCA programming model to expose services that interact with a persistent layer in a declarative fashion hiding the implementation details from the service developer.**

➢ **It's all about simplicity, allowing a service to be defined without explicitly coding the persistence layer.**

Why a Data Service?

➢ **Today the logic for data access is embedded in a service.**

- Logic needs to change when the end source type changes.
- Developer needs to know about APIs that are specific to each end source type.
- Different DAS is needed when using DAS in different language environments.

➢ **This could be simplified by bringing DAS and SCA together into Data Services. Data Service is an extension of the SCA programming model (implentation.das). With this :**

- Developer will be able to focus on business logic and not worry about data access APIs.
- SCA components are language-neutral, therefore a single Data Service can be used across different languages.

APACHE
TUSCANY

Using DAS in Tuscany Today

➢ **Interface definition (Java Interface, required)**

```
public interface AccountService {
    public List getAllCustomers();
    public DataObject getCustomerAccount(String accountNumber);
}
```

➢ **DAS Configuration file (required)**

```
<Config ...>
    <ConnectionInfo dataSource="java:comp/env/jdbc/bigbank"/>
    <Command name="getAllCustomers" SQL="select * from CUSTOMERS"  kind="Select"/>
    <Command name="getCustomerAccount" SQL="SELECT accountNumber, accountType, balance FROM accounts where
accountNumber = ?" kind="Select" />
</Config>
```

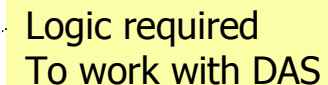➢ **Service Implementation (required)**

```
@Service(AccountService.class)
public class AccountService {
    public List getAllCustomers(){

        DAS das = DAS.FACTORY.createDAS(mapping, conn);
        Command readAllCustomersCommand = das.getCommand("getAllCustomers");
        DataObject root = readAllCustomersCommand.executeQuery();

        // Get a customer object
        List allCustomers = root.getList("CustomerProfileData");
        return allCustomers;
    };
    public Object getCustomerAccount(String accountNumber){   …   };
}
```

Logic required
To work with DAS

APACHE
TUSCANY

Using Data Services in Tuscany

➤ **Interface definition (Java Interface with annotation, required)**

```
public interface AccountService {
    @CommandQuery("SELECT * FROM customers where" )
    public List getAllCustomers();

    @CommandQuery("SELECT accountNumber, accountType, balance FROM accounts where accountNumber = ?" )
    public DataObject getCustomerAccount(String accountNumber);
}
```
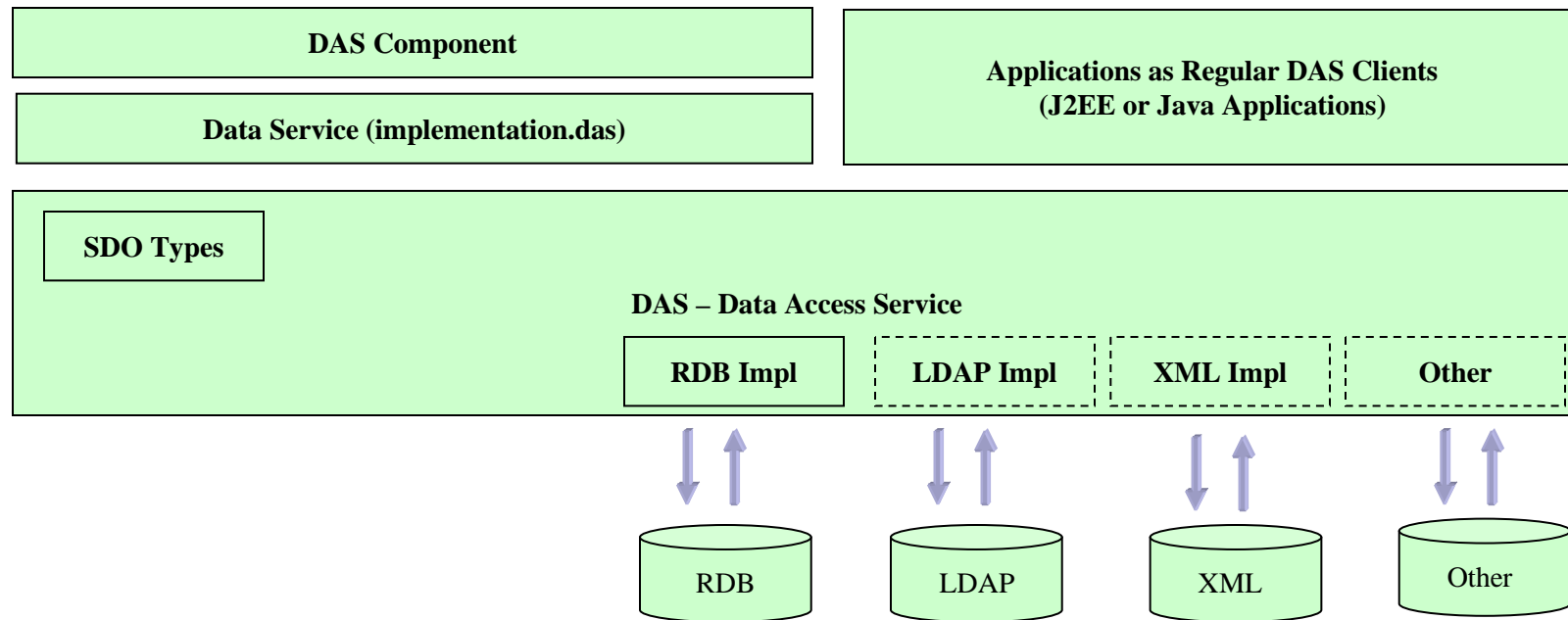
Annotations extends
SCA programming model

➤ **Composite/SCDL file (required)**

```
…
<component name="AccountDataService">
  <interface.java class="bigbank.account.services.account.AccountService "/>
  <implementation.das="dasConfig.properties">
    <connectionInfo  dataSource="java:comp/env/jdbc/bigbank"/>
  </implementation.das>
</component>
…
```

Note: This would also support J2SE environment

High Level Architecture

| DAS Component |
| Data Service (implementation.das) |

| Applications as Regular DAS Clients
(J2EE or Java Applications) |

| SDO Types |

DAS – Data Access Service

| RDB Impl | LDAP Impl | XML Impl | Other |

RDB | LDAP | XML | Other

— **Current DAS**

▪ ▪ ▪ ▪ **Future DAS**

Open Source SOA: SCA, SDO & DAS
http://incubator.apache.org/Tuscany

APACHE
TUSCANY