Service Discovery and Orchestration for Distributed Service Repositories

Ioannis Fikouras
fks@biba.uni-bremen.de
ICSOC03, Trento, 16.12.03

BIBA PPC

Overview

- **Knowledge-Based Variant Configuration**
  - **Object-Oriented Configuration**
- **Constraint Satisfaction**
- **LDAP Distributed Repository**
- **Project NOMAD**
- **Lean Configuration**
  - **Goals**
  - **Principles**
  - **Components**
- **Outlook**

BIBA PPC

Knowledge-Based Variant Configuration (1)

- Knowledge-based Variant Configuration is a process were **complex products are composed out of elementary components**

  Tank, W., "Wissensbasiertes Konfigurieren: Ein Überblick", Künstliche Intelligenz (KI7 Heft2), June 1993

- A Configurator is an expert system that supports this process and thereby uses **predefined goals** as well as **expert knowledge** formulated as constraints, functional requirements, predetermined components or other quality criteria.
- The **greatest hurdle** to Variant Configuration is making decisions without necessary information.
- This can lead to a **dysfunctional composition** or simply to a combination that does not conform to user requirements.

  Neumann B., "Configuration expert systems: a case study and tutorial.", In: Artificial Intelligence in Manufacturing, Assembly and Robotics, H. Bunke (Ed.), Oldenbourg, Munich, 1988

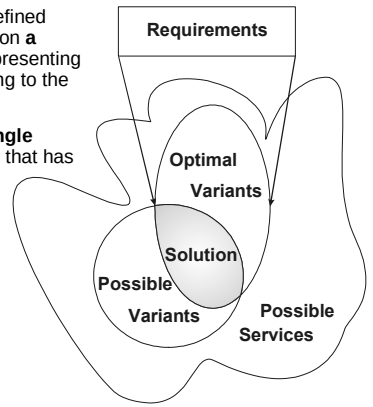
BIBA PPC

Knowledge-Based Variant Configuration (2)

- Configuration does not follow a predefined process, but rather a strategy based on **a series of small steps**, each step representing a certain aspect or assumption leading to the configuration.
- Configuration is the **solution to a single exercise** and not to a whole problem that has first to be methodically analysed.

This implies the following:

1. *The set of all possible solutions is finite.*
2. *The solution sought is not innovative, but rather is a subset of the available parts.*
3. *The configuration problem and the related knowledge domain is known and well defined.*



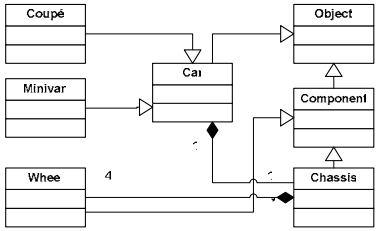
BIBA PPC

Object-Oriented Configuration Principles

- Elementary Components are organised according to a **Product Data Model** into a structure, known as the **Object Hierarchy** that contains all knowledge related to the product in question.
- Product Data Models are **generic** and thus **are not restricted** to a single knowledge- or application-domain.

Cunis R., Günter A., Strecker H. (1991) Begriffshierarchie-orientierte Kontrolle. In: Das PLACON-Buch. Informatik Fachberichte Nr. 266. Springer, Berlin, Heidelberg

- The **Object Hierarchy** contains all relevant objects and the relationships between them in an **"is-a" relationship** that defines:
  - types of objects
  - object classes and subclasses
  - and their properties



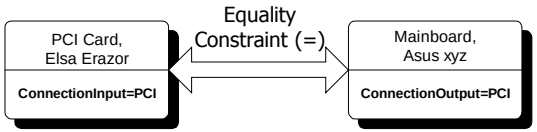
BIBA PPC

Constraint Satisfaction Principles

Relationships between Elementary Components and how they fit together are described with the help of **constraints**.

- Constraints are constructs **connecting** two **unknown or variable components** and their respective attributes, which have predefined values (taken from a specific knowledge domain).
- Constraints connect components by **defining the values** variables are allowed to have as well as the relationship between the two values.
- Constraints contain **general rules** that can be applied to make sure that specific components are put together in a correct fashion **without having to specify any component-related rules** or calculations.

Tsang, E.P.K., Foundations of Constraint Satisfaction, Academic Press, London and San Diego, 1993

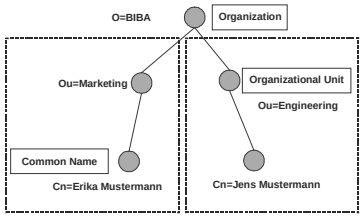


BIBA PPC

LDAP Distributed Service Repository

- LDAP entries are arranged in a **hierarchical** tree structure called the **Directory Information Tree** (DIT).
- Entries referenced by distinguished names constructed out of the Relative Distinguished Name and the names of its ancestor entries.
- Different LDAP nodes can host/be **responsible for different braches** of the DIT.

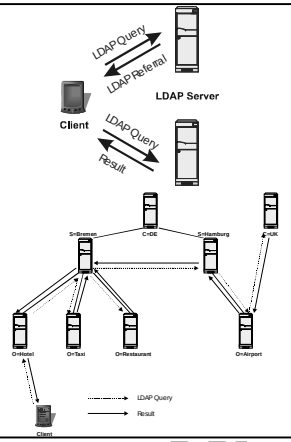
M. Wahl, T. Howes, S. Kille, "Lightweight Directory Access Protocol (v3)" RFC2251-2256,2829-2831, December 1997



BIBA PPC

LDAP Referrals

- When data on the requested services is not available on the local LDAP node (i.e. services registered on remote nodes), the server either:
  - **responds with a pointer** to where the client can get additional information (i.e. another LDAP server)
  - or **initiates further queries** inside the LDAP DIT in order to procure the requested information and then finally delivers it to the client.
- These are respectively **passive** and **active** LDAP referrals.

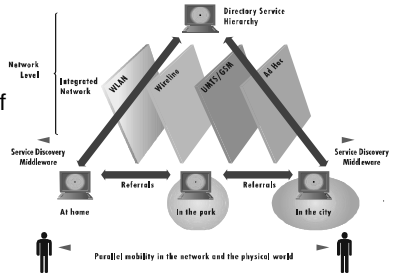


BIBA PPC

NOMAD IST-2001-33292

- Global Roaming through Integration **integration** of **wireless, wireline** and **ad-hoc** access technologies into one network platform
- Transparent **discovery**, **composition** and provision of **context-aware services** on the integrated network platform
- **Distributed service repository** providing Service Discovery and Composition facilities.

http://www.ist-nomad.org



BIBAPPC

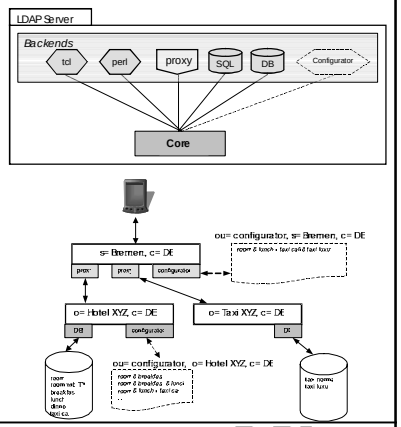
NOMAD Consortium

Germany	Greece	Finland	Switzerland	
T-Systems	MARAC ELECTRONICS S.A.			Industrial
		TEMAGON	TELSCOM	Consultant
BIBA ComNets		HELSINKI UNIVERSITY OF TECHNOLOGY		Academic

BIBAPPC

Distributed Directory & Configurator Engine Integration

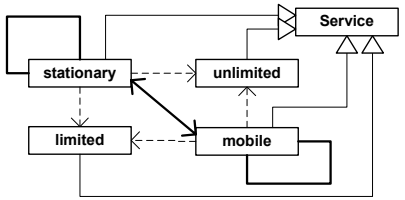
- Implementation of the NOMAD Configurator as an **OpenLDAP** database **back-end**
- Seamless integration with LDAP directory
- **Recursive** usage of Configurator
- Virtual part of the DIT with separate branches for:
  - Services
    - Interfaces
  - Sessions
    - user profiles
    - composite services
    - connections



BIBAPPC

NOMAD Mobile Services Taxonomy

- The basis for **defining elementary services** and the relationships between them in the NOMAD product data model is a provided by a **service taxonomy for mobility aware composite services**.
- This categorization is achieved based on the functionality of the services:
  - **Stationary** or **Mobile**
  - Of **limited** or **unlimited availability**
  - **Information services**



BIBAPPC

Lean Configuration Goals

A Variant Configuration approach based on **requirements** from **Internet applications** (Composition Services for eCommerce & mCommerce Service Platforms) as opposed to industrial product development ERP & PDM platforms.

- **Knowledge-domain independence**
- **Workflow integration**
- **High performance under high load (scalability, modularity)**
- **Interactivity**
- **Soft criteria**

Detken K.-O., Fikouras I. (2000) "Intelligent and secure 3d-configuration of products in electronic shop systems", In: Proceedings of the Third International Conference on Telecommunications and Electronic Commerce (ICTEC3), Dallas, Texas, USA

Fikouras, I., Wunram, M., Weber, F., "Seamless Integration of Mobile Products and Services – User-centricity and Mobility Awareness for mCommerce", In: Proceedings of the Wireless World Research Forum (WWRF) Kick-off meeting, Munich 2001

Pöyry, P., Repokari, L., Fournogerakis, P., Fikouras, I., "User Requirements for Seamless and Transparent Service Discovery", In: Proceedings of eChallenges 2003, 22-24 October 2003, Bologna, Italy

BIBA PPC

Lean Configuration Principles

Object Oriented Variant Configuration and **Constraint Satisfaction** theory based approach

- Usage of **correctly configured, complete compositions** (not specialised or parameterised) as the **basis** for interactive **Configuration**
- Eliminating the complex, computationally intensive and error-prone first two steps of Object Oriented Configuration thereby **eliminating the need for back-tracking**

⇨ Thereby **reducing the Configuration process to a search problem**. Configuration becomes thus the search for the next appropriate component

Fikouras, I., Detken, K., Lean Configuration: Interactive 3D Configuration for E-Commerce Environments, In: J. Gasos, K-D. Thoben (Eds.), "E-Business Applications: Technologies for Tomorrow's Solutions", Springer, Berlin, 2002

BIBA PPC

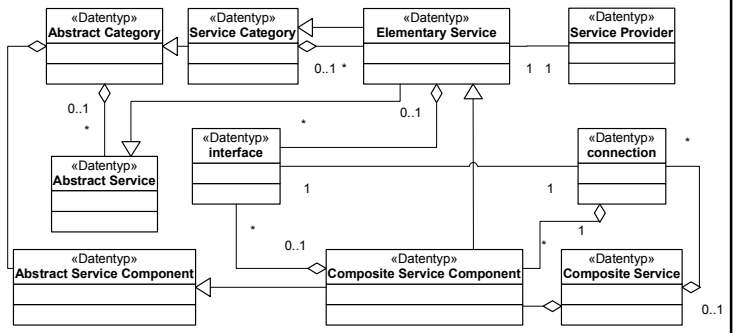
Lean Configuration Knowledge-base Components (1)

A Lean Configuration Knowledge-base consists of the following components:

- **Templates for new configurations** (Composite Services) describing the breakdown of the composition, the workflow and the generic type of components included. Such Composite Service Templates provide the necessary default knowledge for Configuration.
- **Component categories** implementing a means of grouping elementary components into sets according to functional criteria.
- **Elementary components** represent a specific instantiation of a service and contain all data needed to describe it. Elementary components also contain the definition of interfaces available in this service.

BIBA PPC

NOMAD Composite Services Data Model



BIBA PPC

Lean Configuration Knowledge-base Components (2)

A Lean Configuration Knowledge-base consists of the following additional components:

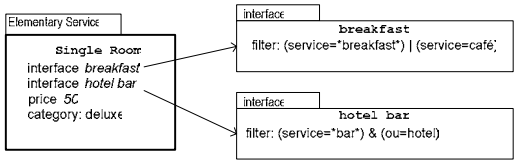
- **Interfaces between elementary components** describing under which conditions co-operation with other components is possible. Such interfaces effectively are **Constraints** enforcing general rules applying to specific interfaces usable by different elementary components.
  - The relationship between interfaces and elementary services matched by the filters contained in an interface resembles the one between **plugs and sockets**, whereby **interfaces as sockets match multiple plugs**.
  - An interface is not restricted in its scope to be used by only one pair of components, but rather **implements a generic rule** (constraint) that can be used by multiple components for describing their interfaces.
- **Connections between elementary components** in a concrete composition describe which components are connected together, by which interface and in what direction for ensuring integrity of the composition.

Fikouras, I., Freiter, E., „Service Discovery and Orchestration for Distributed Service Repositories “, First International Conference on Service Oriented Computing, Trento, Italy, December 2003

BIBAPPC

Elementary Services and Interfaces (1)

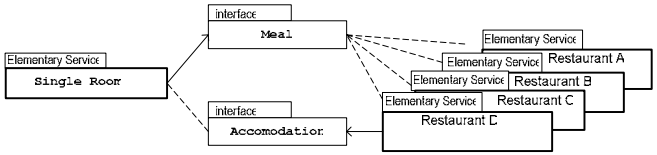
- **Elementary Services** represent a specific instantiation of a service and contain data needed to describe it.
- **Interfaces** between components offer mechanisms for determining whether Elementary Services are suitable for composition into a composite service.



BIBAPPC

Elementary Services and Interfaces (2)

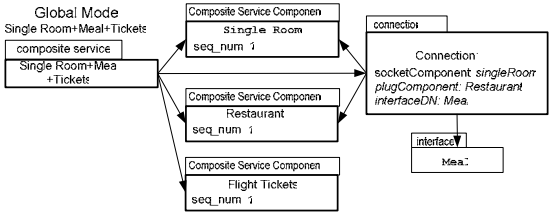
- Interfaces cant not only be used by one pair of components, but rather **implement generic rules** (constraints) that can be used by multiple components.
- Elementary component "socket" (left) with multiple "plugs"(right)



BIBAPPC

NOMAD Composite Services, Connections

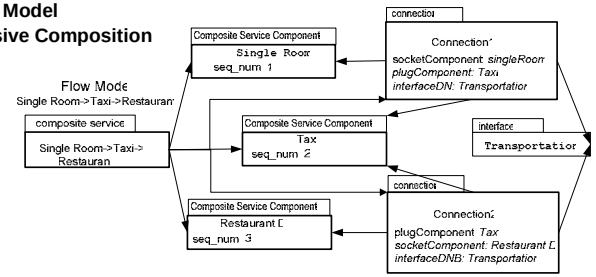
- **Composite Services** consist of groups of **Composite Service Components** derived **individually** from Elementary Services
- When modifying existing Composite Services it is important to know how components are connected to each other (**via which interface**), to avoid altering the structure of the composition.
- Connection components are used for storing information on the connections between composite services and their corresponding interfaces.



BIBAPPC

NOMAD Composite Services & Workflows

- Composite Service Components **inherit their attributes from Elementary Services**, as well as attributes related to workflow management from the Abstract Service Component datatype.
- WSFL compatible workflow models
  - **Flow Model**
  - **Global Model**
  - **Recursive Composition**



BIBA PPC

Outlook

- Current work continues to extend configuration functionality to support context for **context-aware service composition**.
- Context will be derived from **positioning data** as well as static service attributes.
- Measurements & Evaluation
- Integration of a **case-based reasoning** engine for automatically diagnosing what composite service template is appropriate. This would help further automate composition based on Lean Configuration.
- Investigate **semantic aspects** related to dynamic schemata and service taxonomies.

BIBA PPC