

第一届Agent理论与应用学术会议



中科院计算所
INSTITUTE OF COMPUTING TECHNOLOGY

主体网格智能理论和平台 Agent Grid Intelligence Theory and Platform

史忠植

shizz@ics.ict.ac.cn

中国科学院智能信息处理重点实验室



内容提要

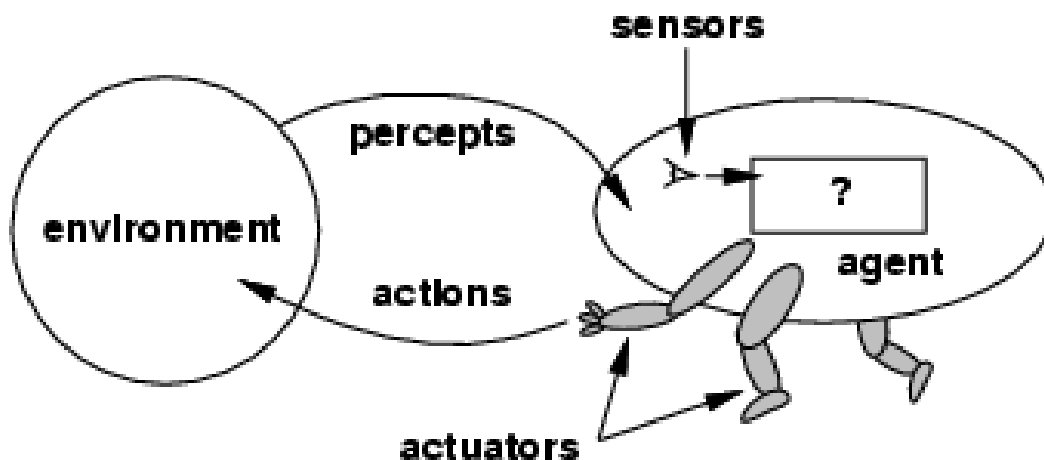
- **概述**
- **智能主体的认知模型**
- **基于主体的SOA**
- **主体协同工作**
- **主体网格智能平台AGrIP**
- **应用**
- **结束语**



主体 (agent)

在计算机和人工智能领域中，主体可以看作是一个自动执行的实体，它通过传感器感知环境，通过效应器作用于环境。

- 自主性
- 交互性
- 协作性
- 反应性
- 进化性





主体的应用

- 语义Web
- 智能机器人
- 先进制造
- 复杂系统建模和仿真
- 群决策支持
- 智能控制
- 协同工作



**2050 Robot team will beat
Person Champion !**



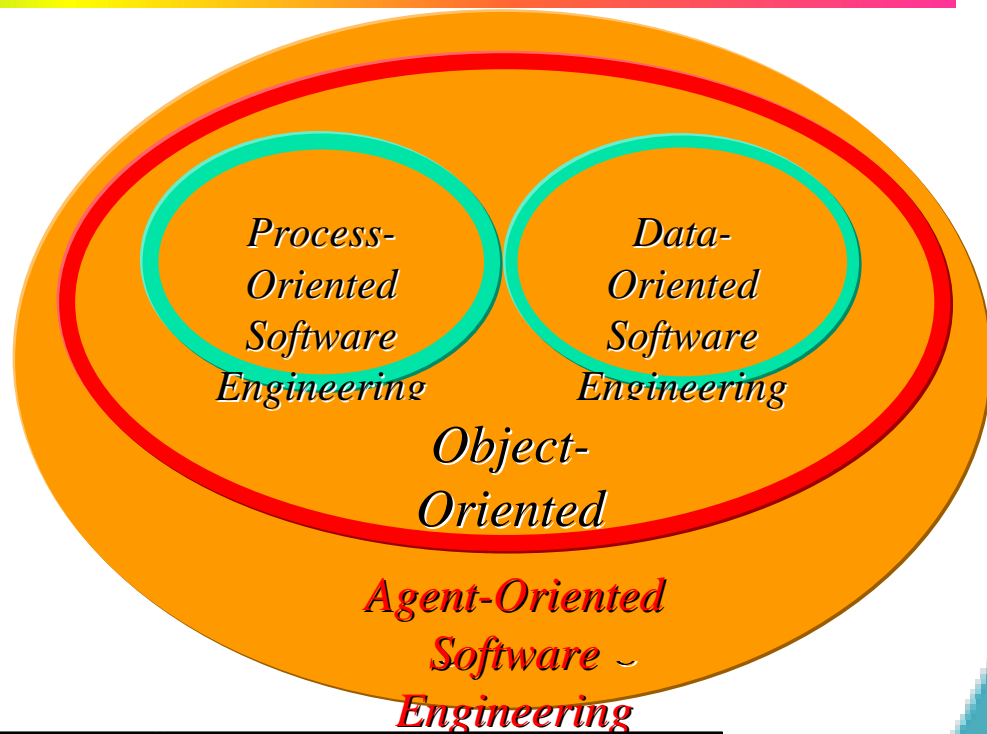
主体技术的发展

- 主体的理论模型-20世纪80年代
 - BDI 模型
 - 思维社会
 - SOAR
- 面向主体的软件工程-20世纪90年代
 - Shoham AOP
 - Wooldridge等: Gaia
 - Agent UML
 - CARBA
- 主体语义网络-21世纪初
 - Tim Berners-Lee: 语义Web
 - 主体网络
 - 主体语义网络



主体技术

主体 (agent) 是一个粒度更大、智能性更高、具有一定自主性的软件实体。主体的自主性、协作性、反应性、演化性、多态性、智能性等特点, 使主体成为理想的整体服务协作软件的基本软件实体。



	面向对象的程序设计 (OOP)	面向主体的程序设计 (AOP)
粒度	细粒度	粗粒度
封装	状态封装	行为封装
服务请求方式	方法请求	服务查询
信念改变方式	内部状态直接操作	知识查询和操作
服务方式	同步服务	异步服务
接口	语法接口	更为自然的语义级接口
继承性	有	一般没有



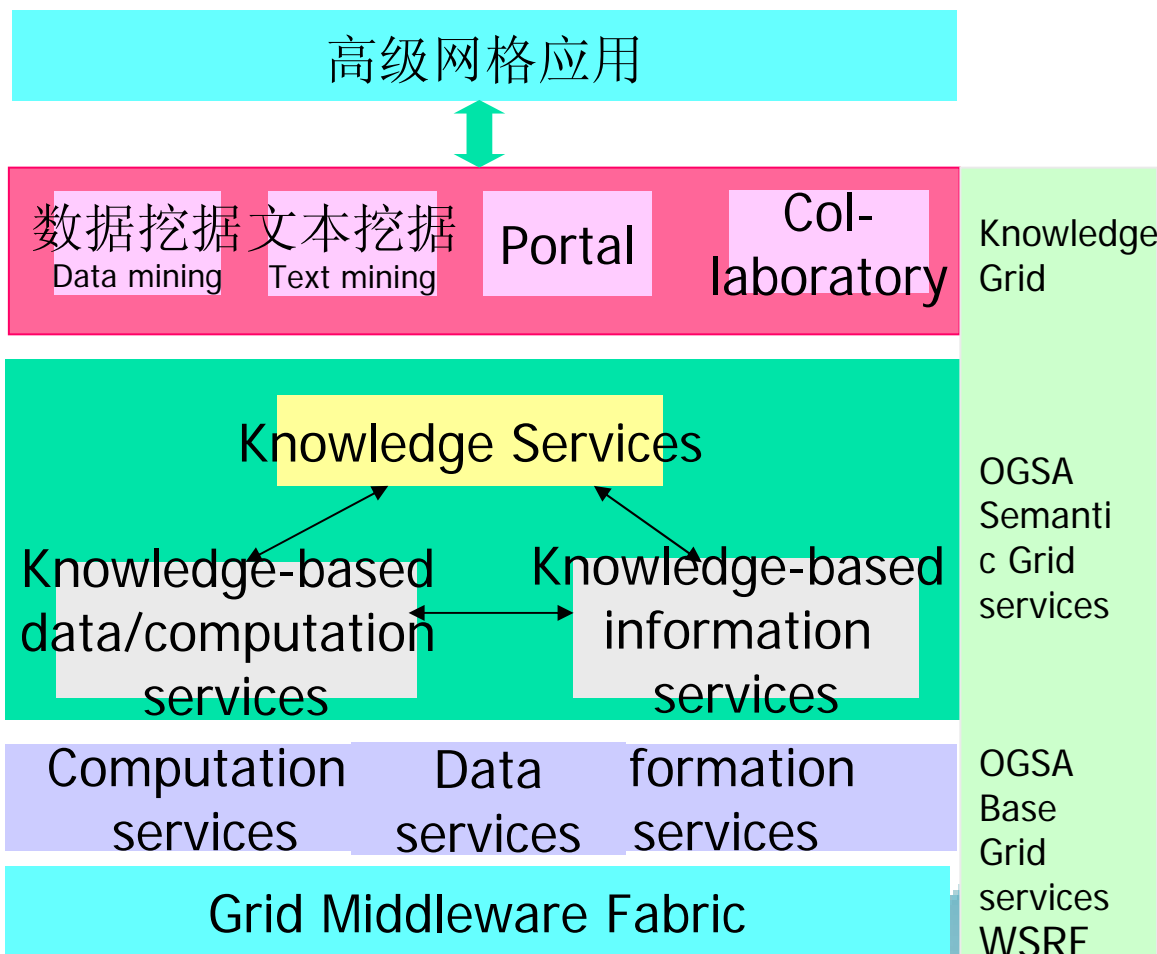
信息技术的新特点

- 自主性—系统实体的主动行为能力
- 协同性—系统实体间协同工作
- 反应性—系统实体的环境感知能力
- 进化性—系统的环境演变能力
- 可信性—系统的可信程度



网格技术

网格把整个互联网集成为一台巨大的超级计算机，实现全球范围的计算资源、存储资源、数据资源、信息资源、知识资源、专家资源、设备资源的全面共享。网格的根本特征是资源共享，消除资源孤岛。





主体技术的研究

重要历程

1994年：多主体处理环境MAPE

1996年：基于主体的计算机支持协同工作环境； RAO逻辑

1998年：面向主体的软件开发环境，提出公共主体请求代理体系机制CARBA，知识信息处理技术和方法 -----知识信息系统环境KISE，荣获1998年中科院科技进步二等奖

2000年：多主体环境MAGE，数据挖掘工具MSMiner

2001年：海洋渔业遥感、地理信息系统技术应用服务系统，荣获2001年中科院科技进步二等奖，2002年国家科技进步二等奖

2002年：提出主体网格智能平台AGr IP系统结构，智能搜索引擎GHunt

2003年：提出动态描述逻辑DDL；

2004年：提出主体服务描述语言SDLSIN

2005年：研制主体网格智能平台AGr IP



内容提要

- **概述**
- **智能主体的认知模型**
- **基于主体的SOA**
- **基于主体的协同工作**
- **主体网格智能平台AGrIP**
- **应用**
- **结束语**

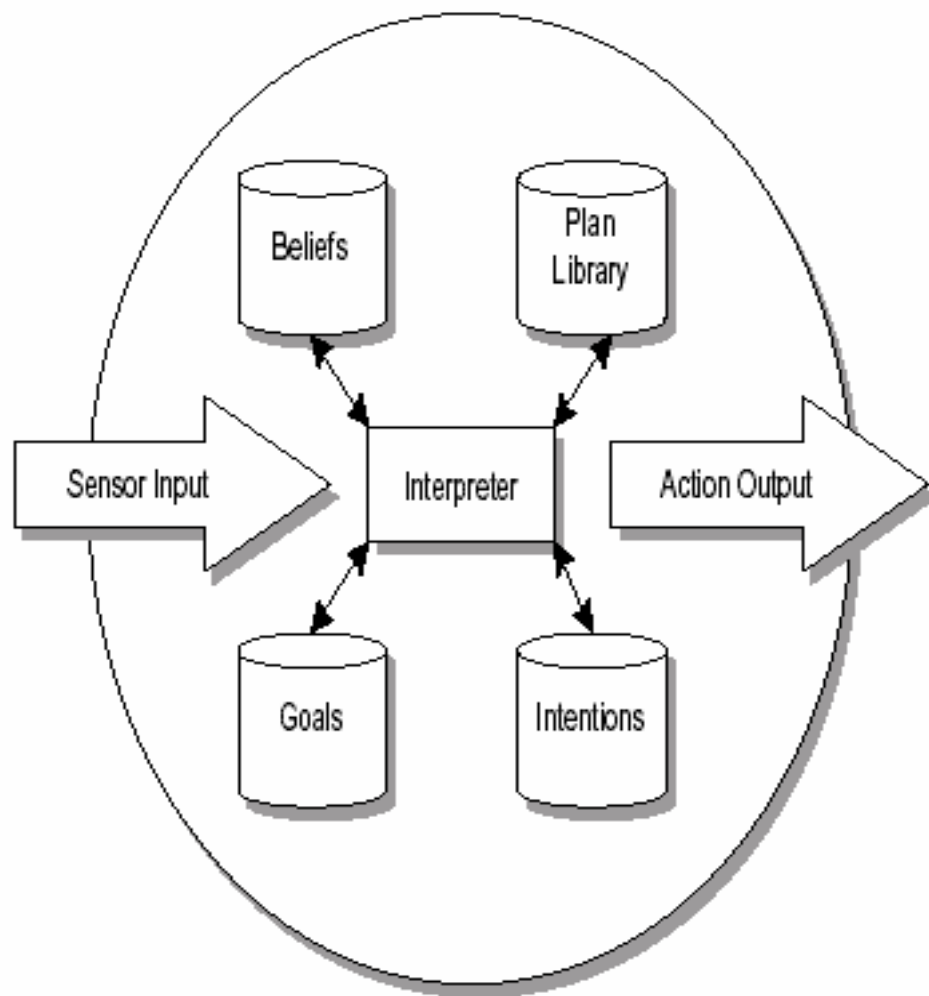


理性主体 (BDI主体)

Belief——信念，
主体对环境的基本
看法。

Desire——愿望，
主体想要实现的状态，
即目标。

Intention——意
图，目标的子集。





主体理论模型

- 1987年Bratman从哲学上对行为意图的研究对人工智能产生了广泛的影响
- Cohen和Levesque基于正规模态逻辑NML的可能世界模型对BDI进行描述
- Rao和Georgeff对BDI模型的描述同样基于正规模态逻辑NML的可能世界模型，但每个可能世界具有分支时间结构
- 石纯一、王学军等的主体博弈模型
- 田启家、史忠植的RAO逻辑主体模型



描述逻辑

TBox语言 是描述领域结构的公理的集合

定义: 引入概念的名称

包含: 声明包含关系的公理

概念 ——表示实体(一元谓词, 类)

关系(Roles) ——属性(二元谓词, 关系)

ABox语言 是描述具体情形(实例断言)的公理的集合

概念断言 ——表示一个对象是否属于某个概念

$a:C$ **Student(Tom)**

关系断言 ——表示两个对象是否满足一定的关系

$\langle a,b \rangle :R$



描述逻辑的问题

■ 描述逻辑

优点：可判定、语义特征、有效的推理服务

缺点：

- 对动态领域的描述能力不够
- 很难对动态领域进行建模
- 还不能有效地对静态和动态有机地整合在一起

■ 动作理论与逻辑程序设计

优点：很好地刻画动态变化的特性，动态领域

缺点：

- 在静态知识表示方面较弱，推理能力不强
- 很难跟实际相结合，缺少同具体领域的结合
- 没有系统完整的理论体系，缺少整体的分析和思考



动态描述逻辑DDL

- 以客观世界的形式化建模为研究内容，提出一种统一的形式化框架，同时从静态和动态两个方面来刻画世界的模型。为动态领域的建模提供一种有效的形式化理论工具；
- 将世界中的概念、关系和个体，公理、事实和动作有机地整合在一起，形成统一的关联和运作机制，使动作与公理和事实相互作用，统一表示并进行推理；
- 对概念、公式和动作都给出了统一的、合适的语法和语义；
- 主体与本体的结合



动态描述逻辑DDL

DDL = DL + Action

- DDL: 语义环境中对服务过程方面的建模和推理
 - 服务改变环境的状态: 前提条件和执行结果, Action
 - 服务描述: 基于World Model, e.g., 描述服务的前提和结果的词汇来自World Model
 - TBox描述World Model的结构
 - 所有状态的描述以数据库或文本的形式分散在Web中
 - 用户或Agent通过查询的方式获得状态的信息, 存放于各自的ABox中
- DL与Action的交互体现在:
 - Action采用TBox中的词汇描述前提和结果
 - Action的执行导致ABox的变化



动态描述逻辑DDL

- **Zhongzhi Shi, Mingkai Dong, Yuncheng Jiang, Haijun Zhang, A Logic Foundation for the Semantic Web. Science in China, Series F Information Sciences, 48(2): 161-178, 2005**
- **W3C中国办事处成立大会暨首届WWW技术研讨会于2006年4月23-24日成功召开。在这次Web技术盛宴上，国内外专家们分别带来了精彩而前沿的技术交流。**
 - Overview of W3C, Steve Brat**
 - Overview of Web Development in China, Jinpeng Huai**
 - Logic Foundation and Service for Semantic Web,**
 - Zhongzhi Shi**



Metal State Model

Mental State: $\langle \mathcal{K}, \mathcal{A}, \mathcal{G}, \mathcal{P}, I \rangle$,

Where

- \mathcal{K} *belief*
- \mathcal{A} **action**
- \mathcal{G} *goal*
- \mathcal{P} **plan**
- I **intention**



Belief

$$\mathcal{K} = \langle \mathcal{T}, \mathcal{S}, \mathcal{B} \rangle$$

\mathcal{T} : *Ontology*

\mathcal{S} : *Constraints*

\mathcal{B} : *Current belief*



Belief Revision

```
AddBelief( $F, \mathcal{B}$ ) {  
     $F' \leftarrow \text{Extend}(F)$ ;  
    Foreach  $\varphi \in F'$  do  
        If  $\neg \varphi \in \mathcal{B}$  Then  $\mathcal{B} \leftarrow \mathcal{B} - \{\neg \varphi\}$ ;  
     $\mathcal{B}' \leftarrow \text{Extend}(\mathcal{B} \cup F)$ ;  
    If Consistent( $\mathcal{B}'$ ) Then Return  $\mathcal{B}'$  ;  
    Else{  
        Let  $\{\psi, \neg \psi\} = \text{ConflictSet}(\mathcal{B}')$  ;  
        If  $\psi \in \mathcal{B}$  Then Return  $\mathcal{B}' - \{\psi\}$ ;  
  
        Else If  $\neg \psi \in \mathcal{B}$  Then Return  $\mathcal{B}' - \{\neg \psi\}$ ;  
        Else Return error;  
    }  
}
```



Goal

Let \mathcal{A} be a set of actions, \mathcal{L} be a set of assertions, \mathcal{G} can be defined recursively:

- (1) $\mathcal{A} \subseteq \mathcal{G}$, \mathcal{A} is basic action;
- (2) If $\varphi \in \mathcal{L}$, then $achieve(\varphi) \in \mathcal{G}$;
- (3) If $\varphi \in \mathcal{L}$, then $\varphi? \in \mathcal{G}$;
- (4) If $\delta_1, \delta_2 \in \mathcal{G}$,
then $\delta_1; \delta_2 \in \mathcal{G}$, $\delta_1 \cup \delta_2 \in \mathcal{G}$, $\delta_1^* \in \mathcal{G}$;



Static Plan

$$\delta_1 \Leftarrow \varphi \mid \delta_2$$

$$\delta_1 \in \mathcal{G}, \quad \delta_2 \in \mathcal{G}, \quad \varphi \in \mathcal{L}.$$

δ_1 : Rule header

δ_2 : Rule body

φ : Rule guard condition

$go_floor(x) \Leftarrow @At(elevator, y) \mid [y < x?; up(x) \cup y > x?; down(x)]$



Dynamic Plan

Plan(δ, B) //Goal δ , B Current belief set

{

If perform action α $B \vdash T^\delta$ Then

{ Return B ; *Enqueue*(α, P); }

Else {

look for subgoals to reach goal ;

computing E_α and δ for each subgoal;

computing the order of subgoals

select optimal subgoal plan;

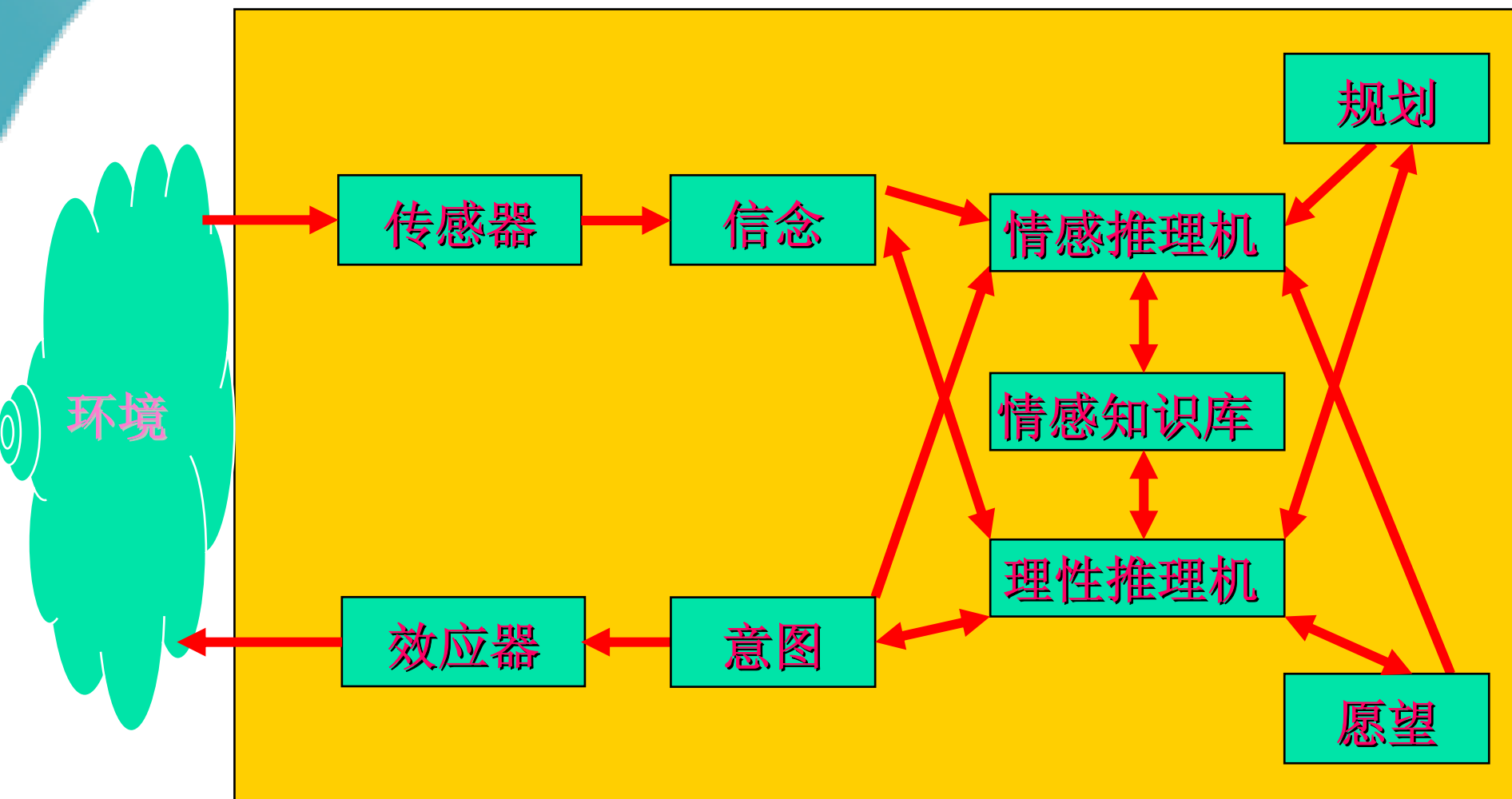
sorting subgoals and form sequence $\delta_1, \dots, \delta_n$

For $\pi = \delta_1$ to δ_n do

$B = \text{Plan}(\pi, B)$



情感智能主体结构



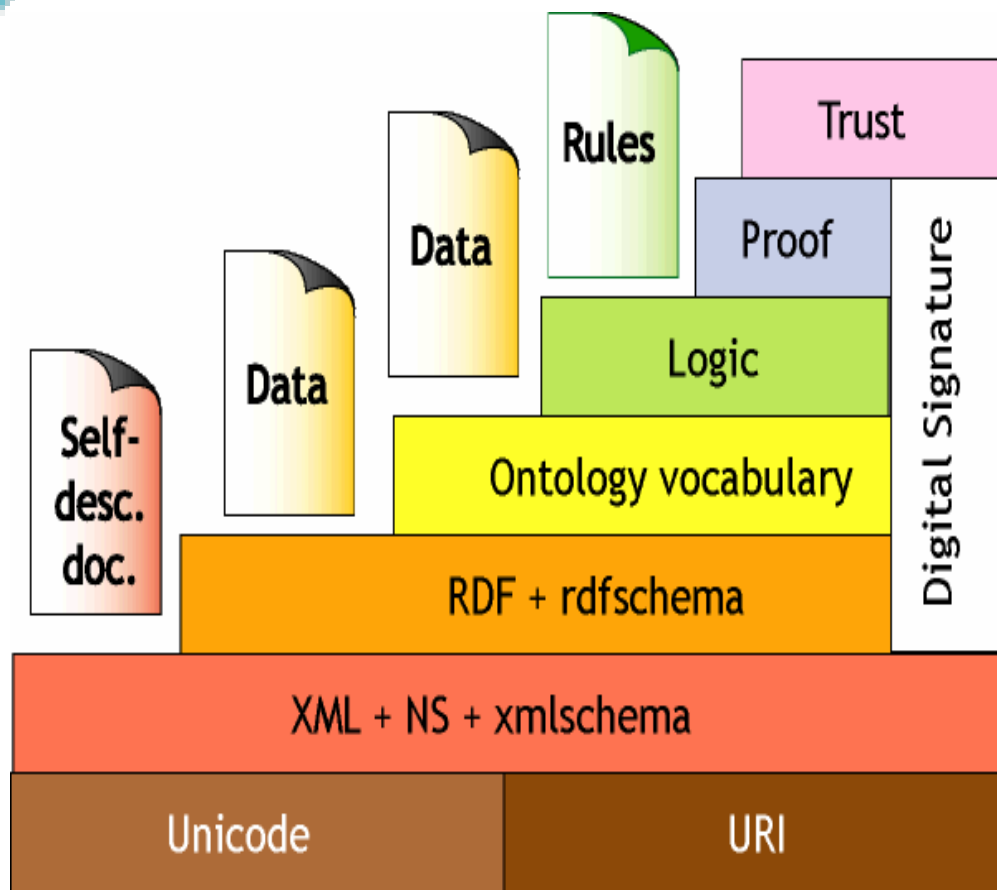


内容提要

- 概述
- 智能主体的认知模型
- 基于主体的SOA
- 基于主体的协同工作
- 主体网格智能平台AGrIP
- 应用
- 结束语



Web语言层次



■ 10 Feb 2004

W3C

Recommendation

OWL

by Tim Berners-Lee



W3C规范

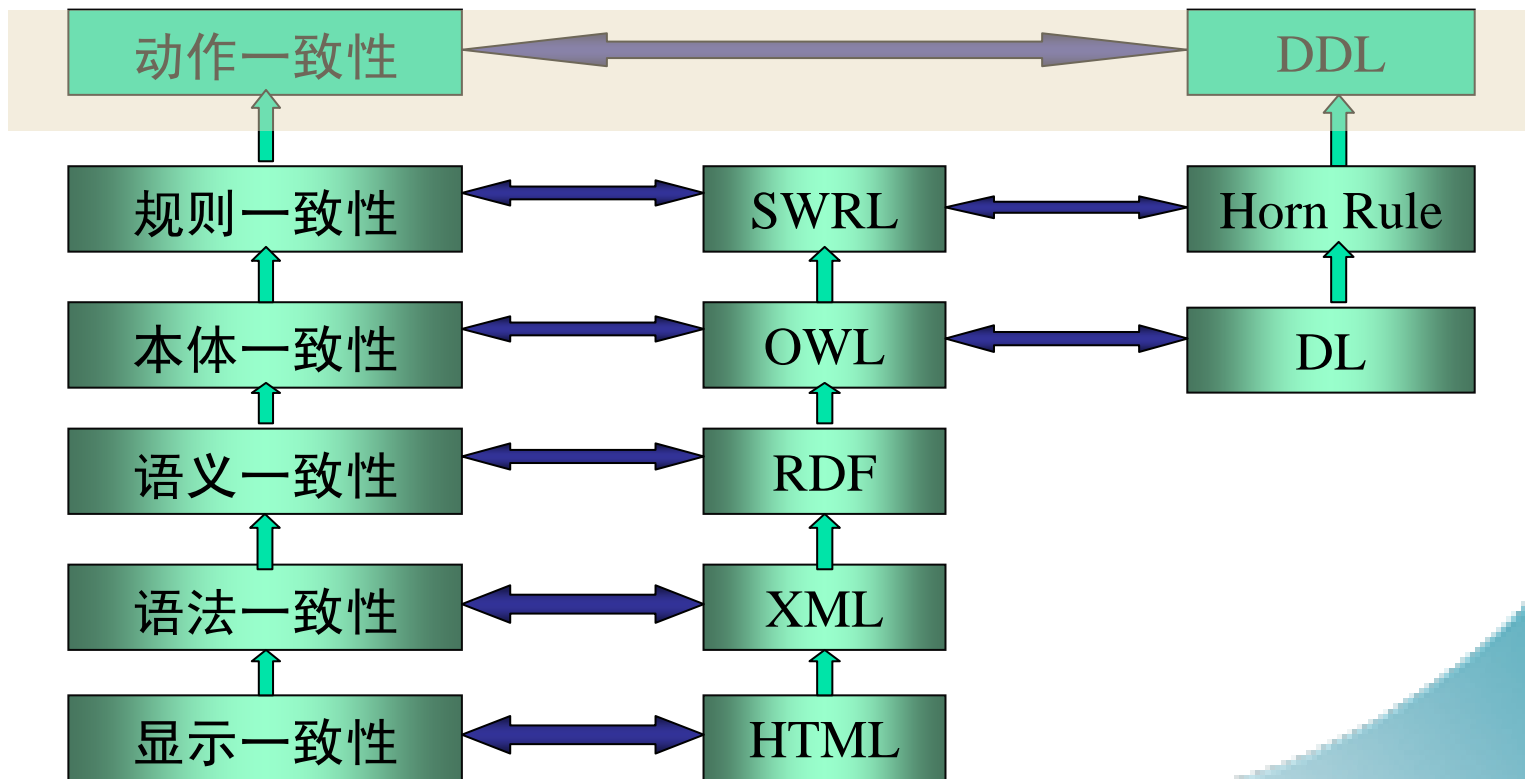
语义Web - - 下一代Web

- 基于语义内容的推理
- 面向服务的过程推理

描述逻辑

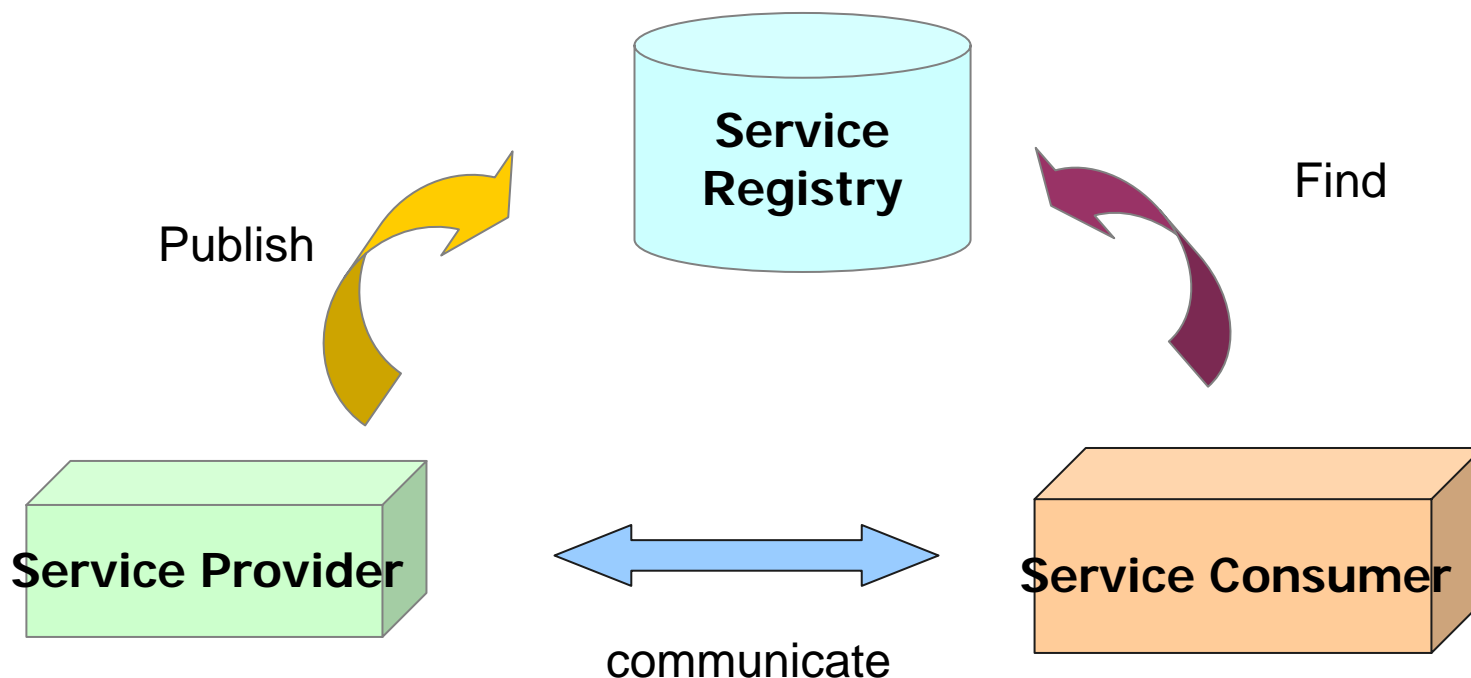
- 为静态知识的表示, e.g., OWL-S
- 不能对服务的动态性进行推理

W3C
|
现有的规范





Web 服务结构



- Web Service : 提供了一个松散耦合的集成模型以支持全球范围内的一致化的互操作。



语义Web服务

Bringing the web to its full potential

Dynamic

Web Services

UDDI, WSDL, SOAP



Semantic Web Services



Static

WWW

URI, HTML, HTTP



Semantic Web

RDF, RDF(S), OWL



语义Web服务

语义Web技术

- 机器理解数据
- 数据模型是本体

+

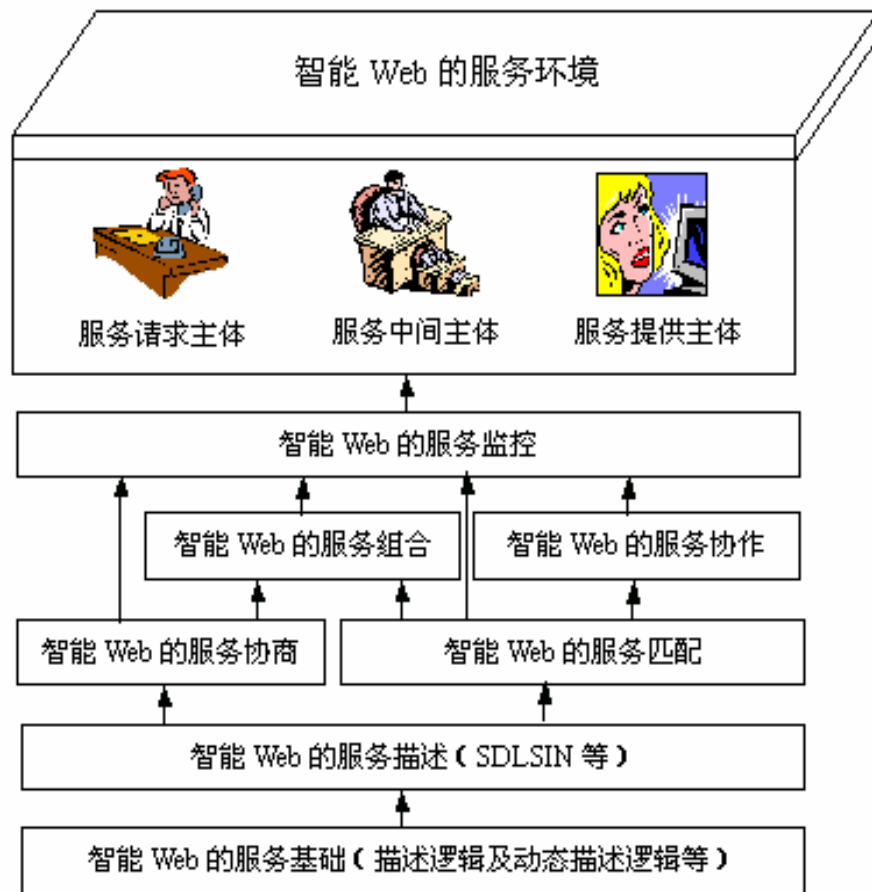
Web 服务技术

- 自动发现、选择和组合
- 基于Web实现服务

=> 语义 Web 服务是下一代Web 的集成解决方案

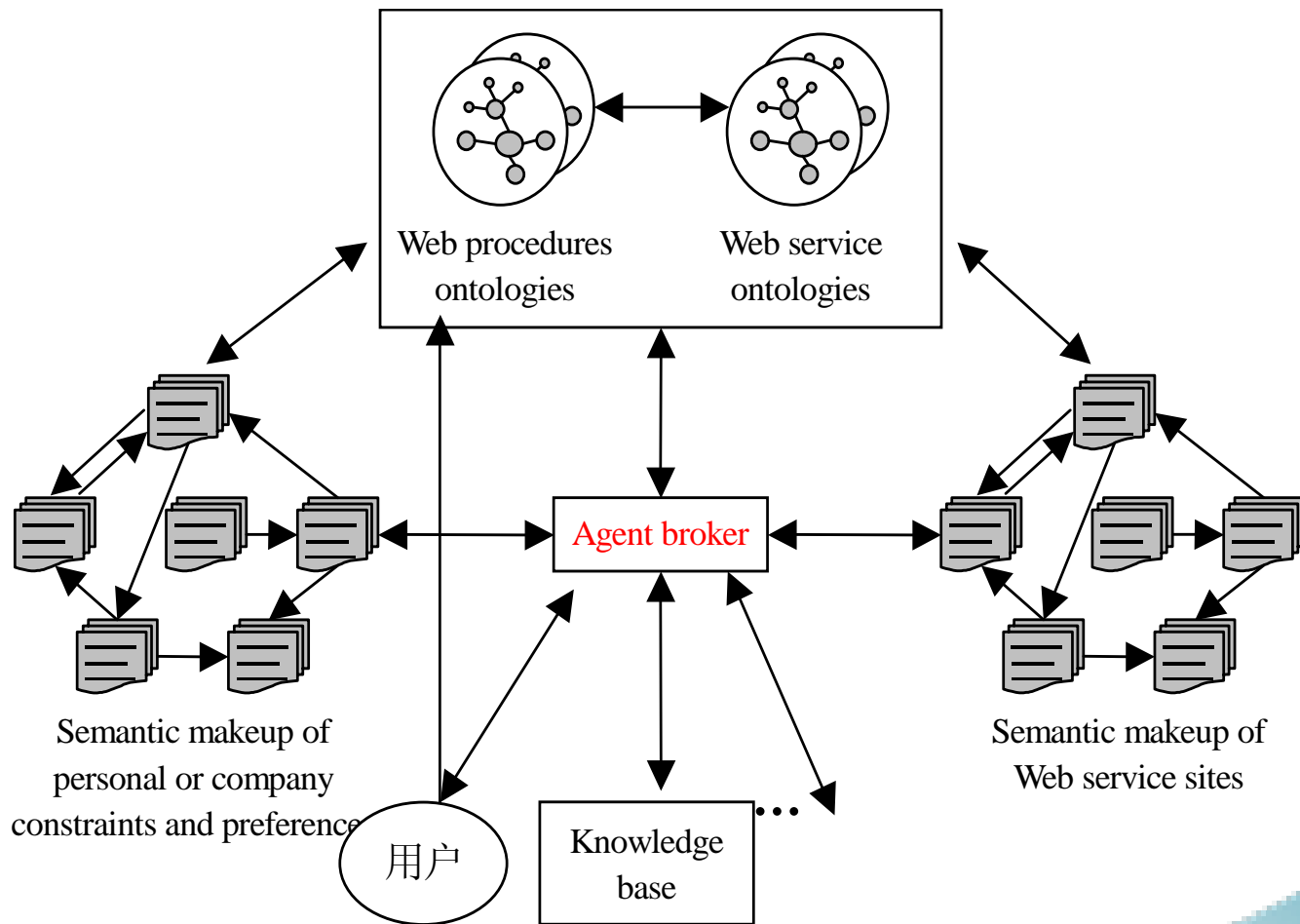


智能Web的服务架构





基于主体的语义Web服务组合结构





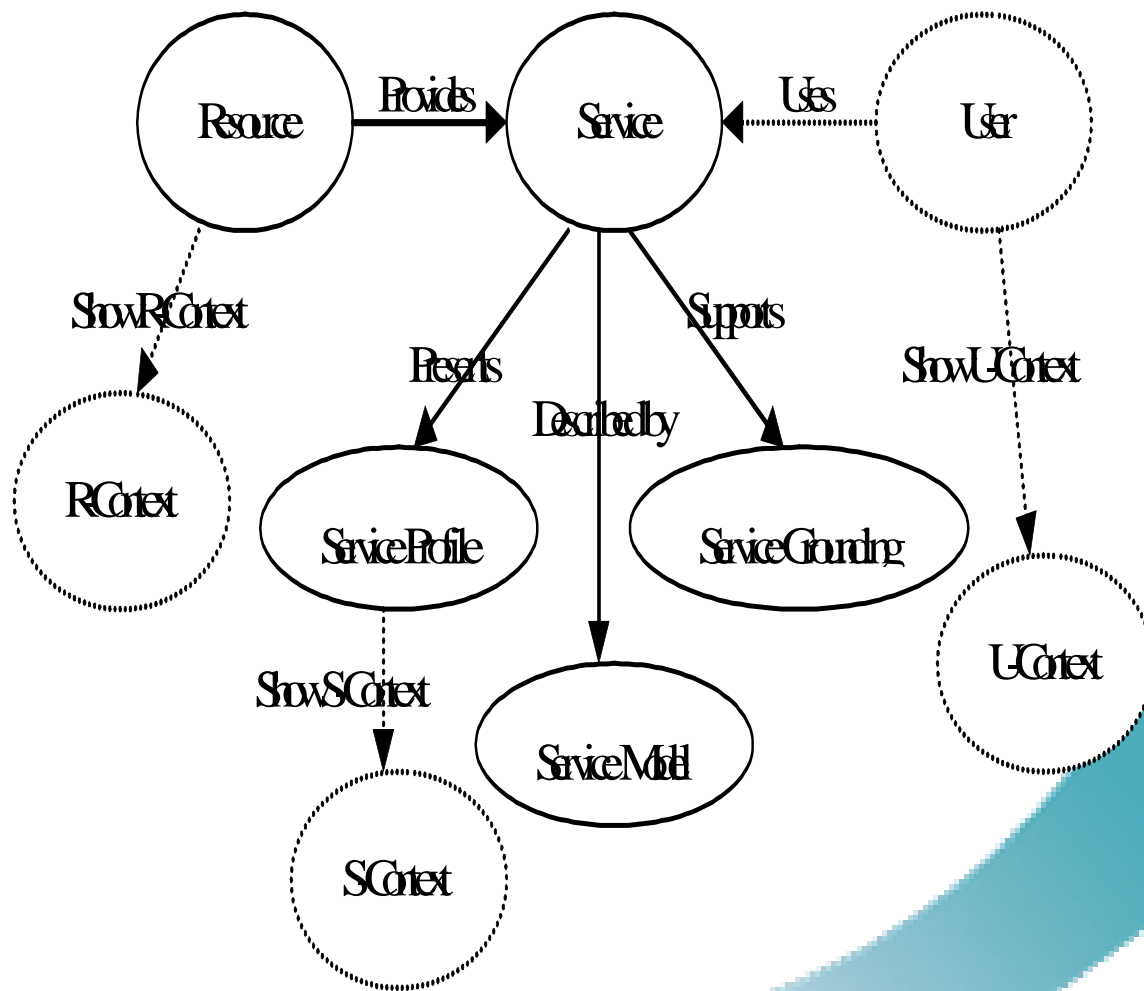
基于主体的语义Web服务组合结构

- 主体要能根据服务请求者的个人偏好和请求限定动态发现服务。这就要求主体能够根据服务请求者的语境(Context)信息，动态感知服务的属性。
- Web服务的描述要能使得主体能调用查找到的Web服务。
- 考虑不同粒度的服务组合。



OWL-S语境

- 在Web服务组合中，Context信息分为三部分：
- 用户U-Context记录用户当前状态和个性化信息；
- 服务S-Context记录Web服务的当前状态和执行的限定条件；
- 资源R-Context用来记录当前的资源状态。





语境与Web服务组合

- 扩展后的OWL-SContext模型，不仅包含服务的过程模型，还包含语境信息。
- 服务描述包含Precondition和结果effect，因此把服务看作主体的动作action，主体根据特定目标推导出服务的组合序列，而语境信息的表示可以影响服务序列的选择。
- 基于DDL的语义Web服务形式化描述



主体服务描述语言SDLSIN

❖ 主体服务描述语言SDLSIN

SDLSIN语言是一种带槽结构的框架表示语言，它的形式规范如下：

```
<asdl-descr> ::= (ctype
  :service-name name
  :context context-name+
  :types (type-name = <modifier> type)+
  :isa name+
  :inputs (variable: <modifier> put-type-name)+
  :outputs (variable: <modifier> put-type-name)+
  :input-constraints (constraint)+
  :output-constrains (constraint)+
  :io-constrains (constraint)+
  :concept-description (ontology-name = ontology-body)+
  :state-language name
  :concept-language name
  :attributes (attributes-name : attributes-value)+
  :text-description name
)
```



内容提要

- 概述
- 智能主体的认知模型
- 基于主体的SOA
- **主体协同工作**
- 主体网格智能平台AGrIP
- 应用
- 结束语



主体协同工作

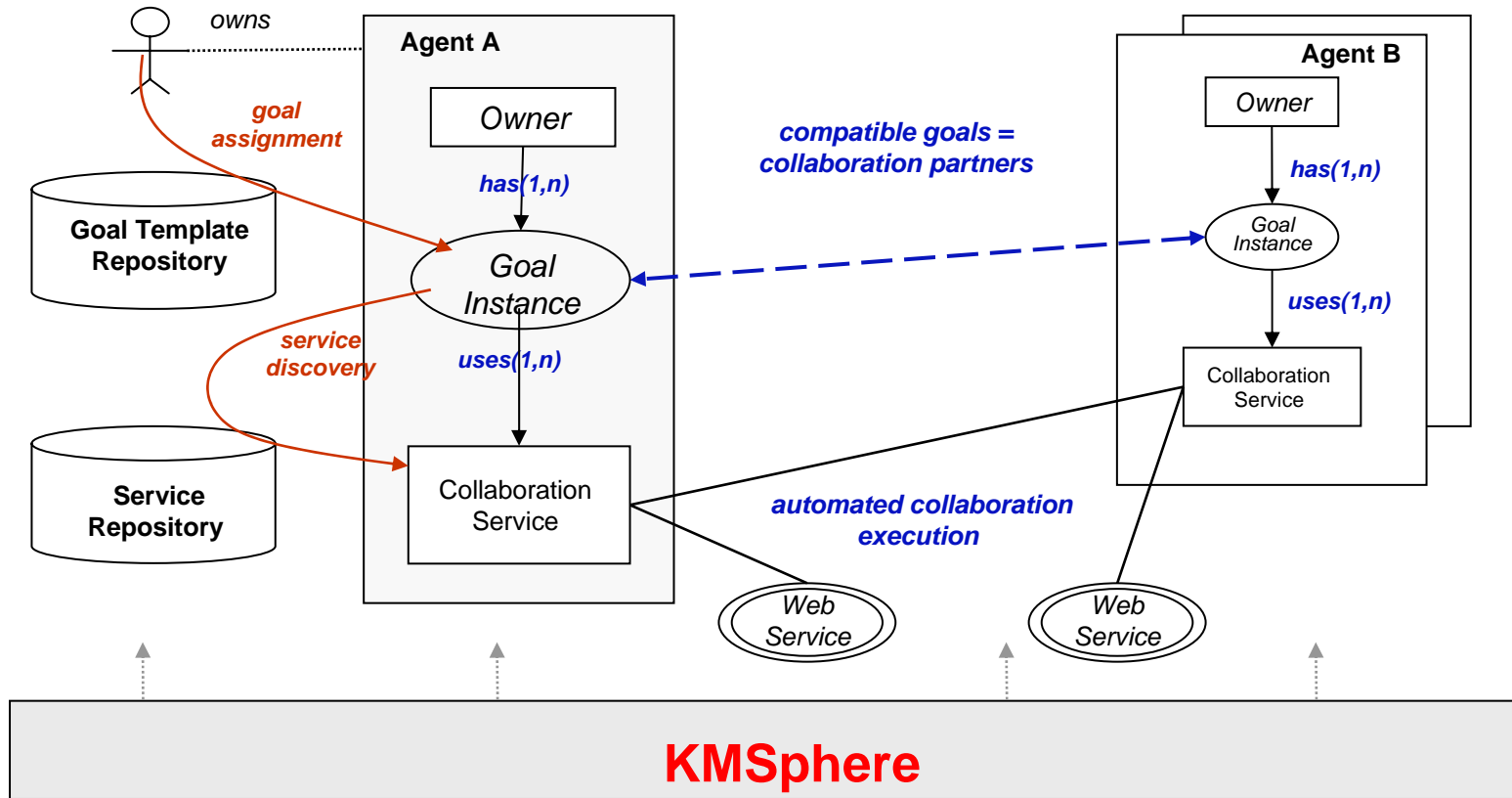
- 主体通讯语言ACL
- workflow
- 基于本体的知识管理
- 策略驱动
- 规划



协同作战

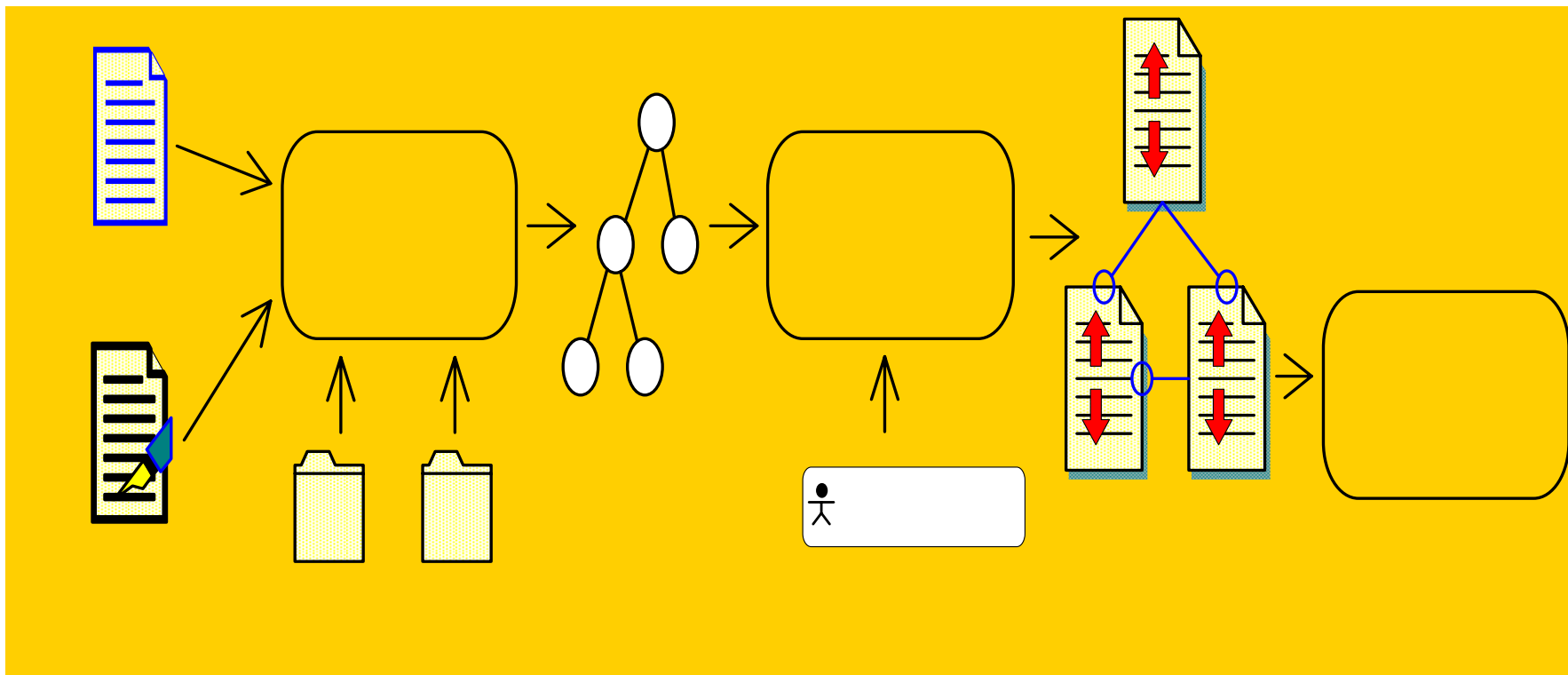


基于知识管理的协同架构





KMSphere

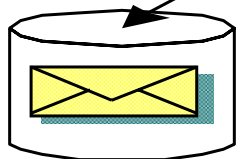
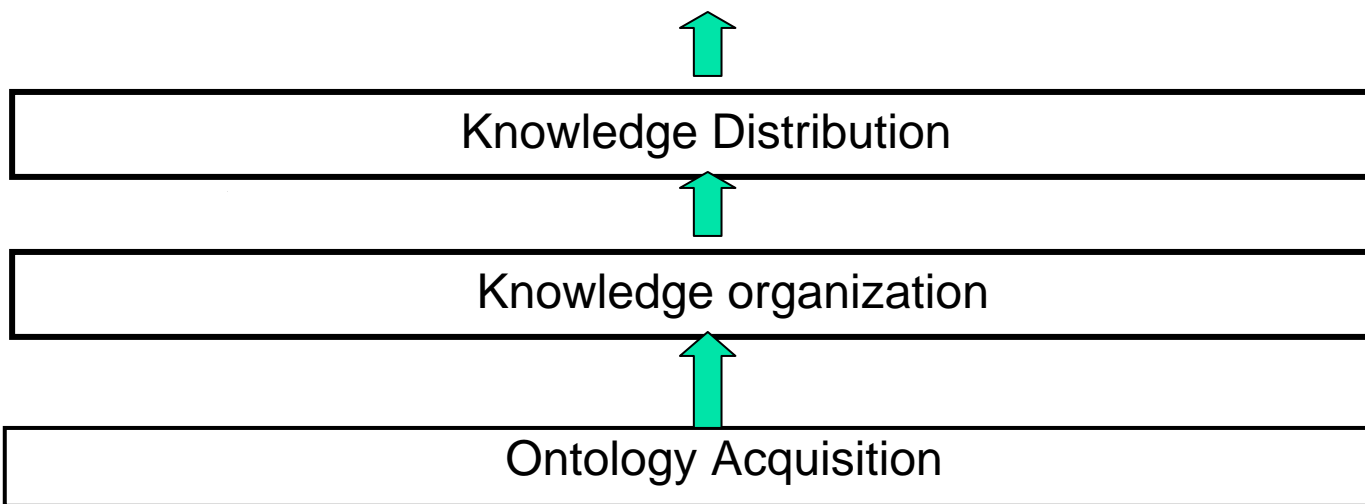




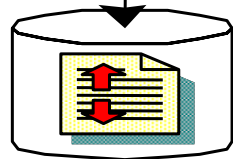
KMSphere



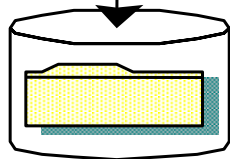
Knowledge Application



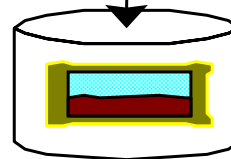
Email



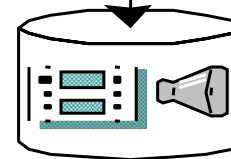
Document



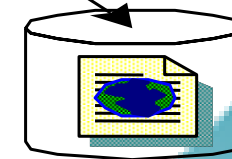
File



Image



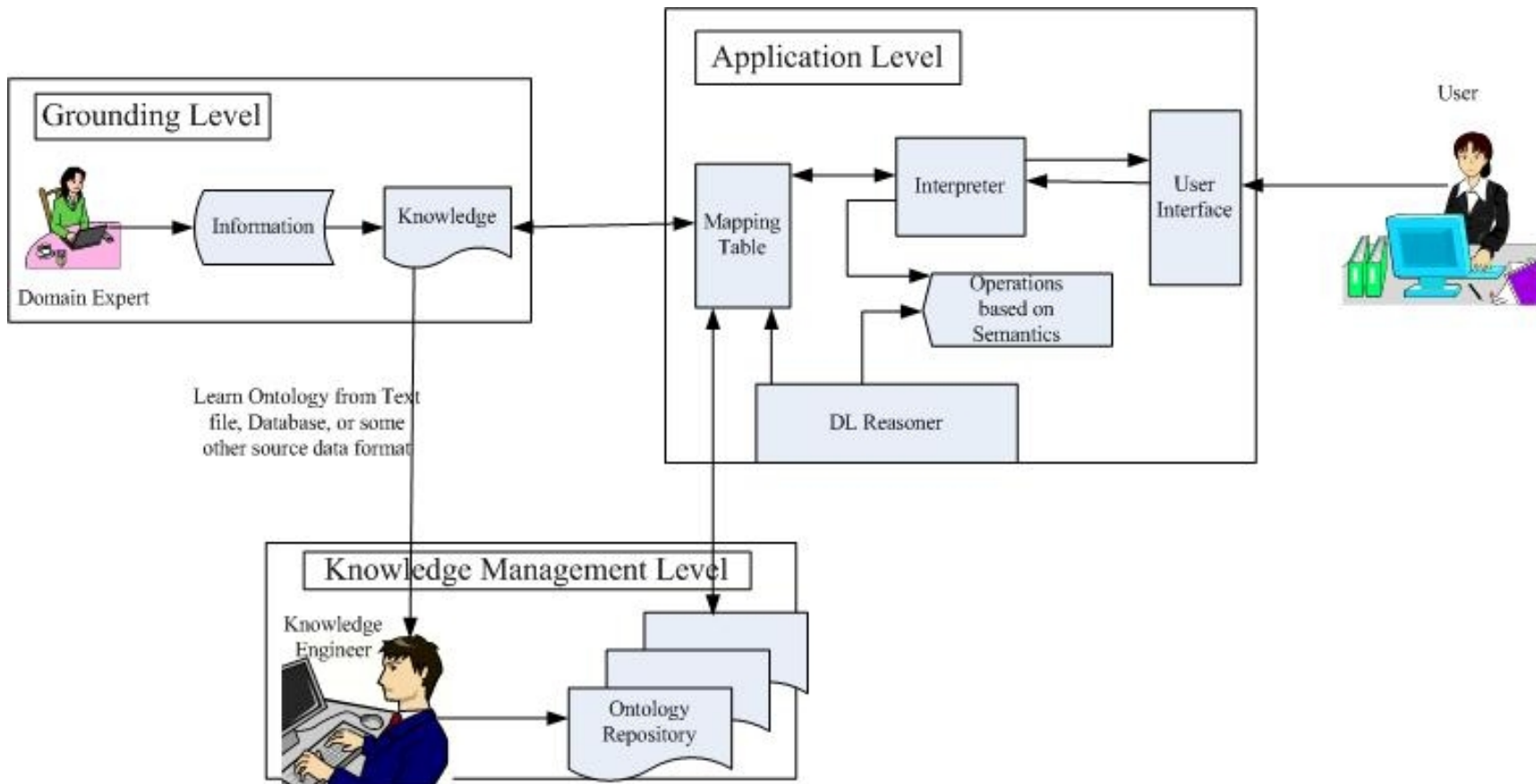
Video



Web



KMSphere





KMSphere Demo

The screenshot shows the KMSphere 3.1 software interface. The main workspace displays an ontology diagram with the following structure:

- Thing (Root Class)
- Food (Subclass of Thing)
- Drink (Subclass of Thing)
- Dessert (Subclass of Food)
- Wine (Subclass of Drink)
- Cake (Subclass of Dessert)

A yellow thought bubble is positioned near the Wine node, containing the text: "Create ontology by hand".

The interface includes a menu bar (File, Edit, Project, Help), a toolbar with various icons, and a left sidebar with sections for "Create Object", "IS-A Relations", "Properties", "Restrictions", "Set Operators", and "Operators". The right sidebar contains fields for "Class Name", "Label", "Namespace", and "Comments".



KMSphere Demo

The screenshot shows the KMSphere 3.1 application window. The title bar reads "food KMSphere 3.1 (http://www.w3.org/2001/sw/WebOnt/guide-src/food)". The menu bar includes "File", "Edit", "Project", and "Help". The toolbar contains various icons for file operations and editing. The main window is titled "数据库本体学习" (Database Ontology Learning) and contains three buttons: "打开数据库" (Open Database), "本体编辑" (Ontology Edit), and "退出" (Exit). The main content area displays information about a database named "CINEMA" with fields: CINEMANAME, CINEMAADDRESS, CINEMATEL, and CINEMAPOST. Below this, it shows information for "ENTERTAINMENT" with fields: ENTERTAINMENTNAME, ENTERTAINMENTADDRESS, ENTERTAINMENTTEL, and ENTERTAINMENTPOST. At the bottom, it shows information for "HOTEL" with fields: HOTELNAME, HOTELADDRESS, HOTELTEL, and HOTELPOST. A "本体编辑" (Ontology Edit) dialog box is open, showing a list of concepts on the left: "影院" (Cinema), "ENTERTAINMENT", and "旅馆" (Hotel). The "数据属性" (Data Properties) column contains "HOTELNAME", "HOTELADDRESS" (highlighted), "HOTELTEL", and "HOTELPOST". The "对象属性" (Object Properties) column is empty. Below the lists are input fields for "旅馆" and "HOTELADDRESS", and a "string" data type field. Buttons for "保存Owl" (Save Owl) and "取消" (Cancel) are at the bottom of the dialog. The footer of the application window reads "中国科学院计算技术研究所智能信息处理重点实验室" (Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences).

Ontology acquisition from databases



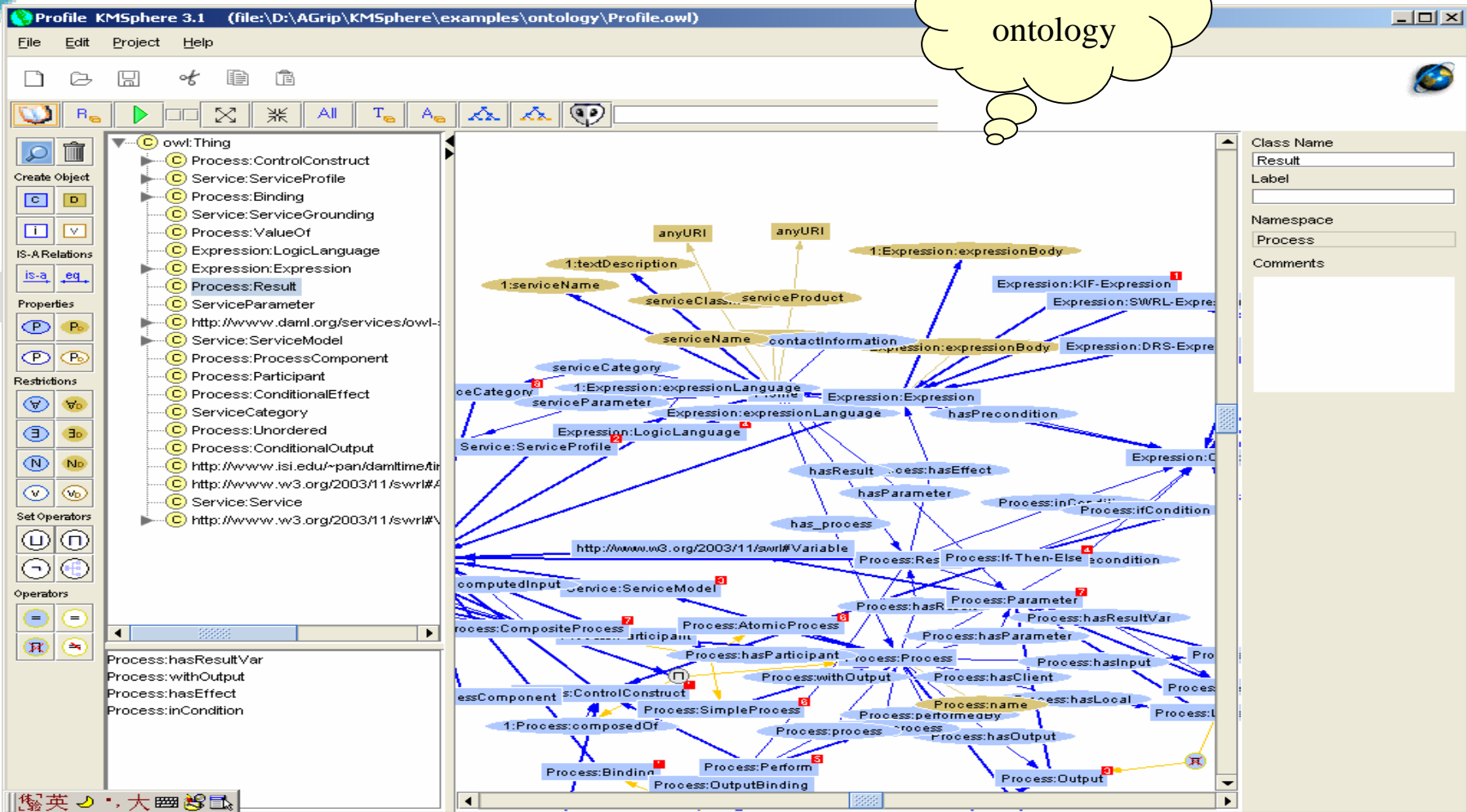
KMSphere Demo

The screenshot displays the KMSphere 3.1 application window. The main interface includes a menu bar (File, Edit, Project, Help), a toolbar with various icons, and a central workspace. On the left, there is a vertical toolbar with sections for 'Create Object', 'IS-A Relations', 'Properties', 'Restrictions', 'Set Operators', and 'Operators'. On the right, a panel shows fields for 'Class Name', 'Label', 'Namespace', 'owl', and 'Comments'. A yellow thought bubble with the text 'Ontology acquisition from text' is positioned in the center-right of the workspace. Two smaller windows are open: one titled '基于文本的半自动本体获取' (Semi-automatic ontology acquisition from text) and another titled '规则' (Rules) containing a table with columns 'Name' and 'Topic'.

Name	Topic



KMSphere Demo





KMSphere Demo

The screenshot displays the KMSphere 3.1 application window. The main area is divided into several panes. On the left, a tree view shows a single class named 'Thing'. The central pane is titled 'KMSphere Consistency Reasoner' and contains a text input field for the URI, which is set to 'D:\AGrip\KMSphere\examples\ontology\pizza.owl'. Below the input field is a 'Run Reasoner' button. The bottom section of this pane displays the 'Reasoner Results' for the input file, showing the following information:

```
Reasoner Results:  
Input file: file:D:\AGrip\KMSphere\examples\ontology\pizza.owl  
OWL Species: DL  
DL Expressivity: ALCF(D)  
Consistent: Yes  
Time: 8156 ms (Loading: 5578 Preprocessing: 0 Species Validation: 2344 Consistency: 31 )
```

On the right side of the application, there is a panel for class details, including fields for 'Class Name' (Thing), 'Label', 'Namespace' (owl), and 'Comments'. A yellow thought bubble with the text 'Ontology consistency check' is overlaid on the bottom right of the interface.

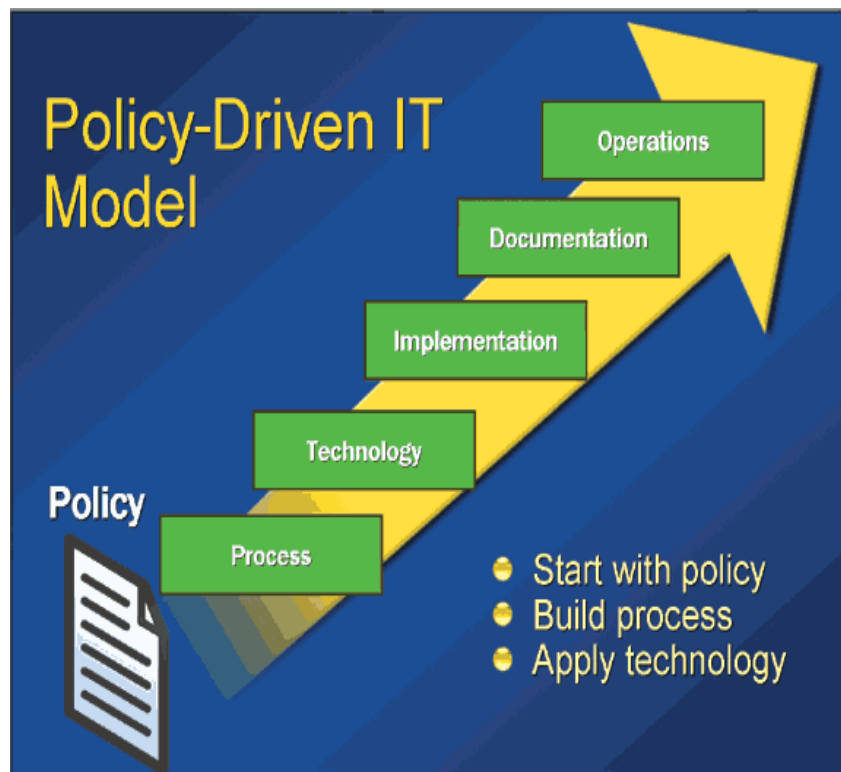


策略驱动

策略是引导系统行为的手段。

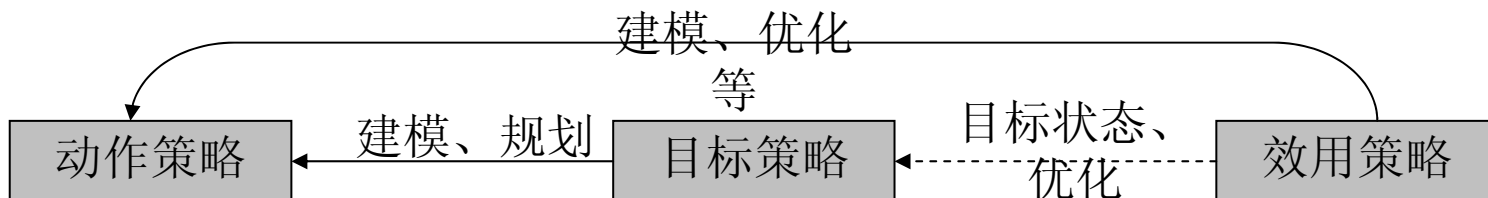
$$p = \langle S_{\text{trigger}}, A, S_{\text{goal}}, U \rangle$$

Strigger: 触发状态集
A: 动作集
Sgoal : 目标集
U: 效用函数集





策略驱动



动作策略:

访问控制策略 (Access Control Policy)

职责策略(Obligation Policy)

目标策略 (Achieve)

描述系统所要达到的目标

效用策略 (Optimize)

使用目标效用函数对每种可能状态作出评估, 选择最优状态



多主体规划

- 规划
 - 规划是主体为达到目标所制定的计划；
 - 规划包括动作集合和动作上的约束集合；
- 多主体规划中特有的问题
 - 不同主体之间的动作冲突；
 - 约束有可能造成死锁。



规划算法

算法 *Distributed – MultiAgent – Plan*(OPS, CaS, I, G)

输入：所有动作模板的集合 OPS ，所有主体的能力集合 CaS ，当前环境状态 I ，目标 G

输出：规划 Π

- 1、通知所有主体规划开始，并等待所有主体执行完当前动作后进入规划状态。
- 2、初始化：建立初始规划 $\Psi = \langle SO, OO, CS, CLS \rangle$ ，令 $SO = \{Init, Goal\}$ ， $OO = \phi$ ， $CS = \phi$ ， $CLS = \phi$ ；将目标 G 分解为子目标集合 GS ，先令 $GS = \phi$ ，然后对每个原子谓词 $g \in G$ ，向 GS 中加入 $\langle g, Goal \rangle$ 。
- 3、调用递归算法 $\varphi = Generate – Next – Step(\Psi, GS)$ 。
- 4、规划生效：对于规划中有其它主体参与的动作以及与其他主体有关的约束，通知相应的主体。
- 5、通知所有主体规划结束，所有主体继续执行自己的规划。
- 6、如果成功，则返回 φ ，否则失败退出。



多主体规划

- 适用于多主体协同工作；
- 是一种分布式规划算法；
- 通过约束传播来检测与消解主体规划之间冲突。



内容提要

- 概述
- 智能主体的认知模型
- 基于主体的SOA
- 主体协同工作
- 主体网格智能平台AGrIP
- 应用
- 结束语



主体网格智能理论

本成果将主体的智能性、自主性和网格的共享性、互操作结合起来，研究网络信息时代的软件理论和方法学，在主体网格软件架构和服务模式基础上，研究智能软件开发方法和工具、基于本体的知识管理、协同工作模式等方面，取得重大进展，建立智能化软件规模生产的开发平台，促进智能软件产业和智能信息处理的发展。

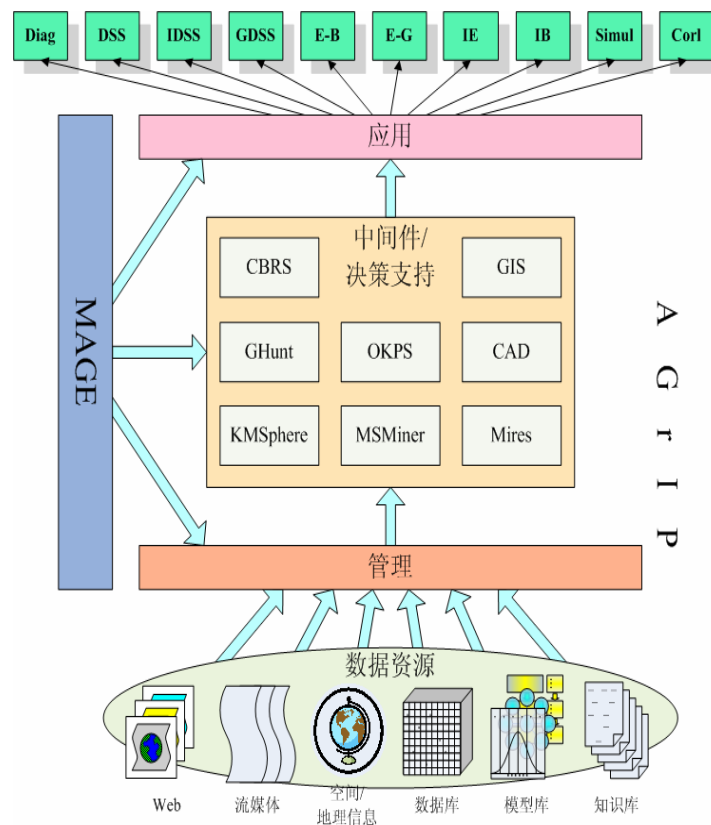
Zhongzhi Shi, Mingkai Dong, Haijun Zhang, Qiu Jian Sheng. [Agent-based Grid Computing](#). Keynote Speech, International Symposium on Distributed Computing and Applications to Business, Engineering and Science, Wuxi, Dec. 16-20, 2002

Zhongzhi Shi, He Huang, Jiewen Luo, Fen Lin, Haijun Zhang. Agent-based Grid Computing. *Applied Mathematical Modeling*, 30(2006): 629-640.

ICT 主体网格智能平台 AGrIP

AGrIP系统结构

- 多主体平台MAGE (Multi-Agent Environment)
- 中间件工具层:
 - 专家系统工具 **OKPS**
 - 智能信息搜索工具 **GHunt**
 - 多策略数据挖掘工具 **MSMiner**
 - 范例推理工具 **CBR**
 - 多媒体检索工具 **MIRES**
 - 知识管理系统 **KMSphere**
- 应用层



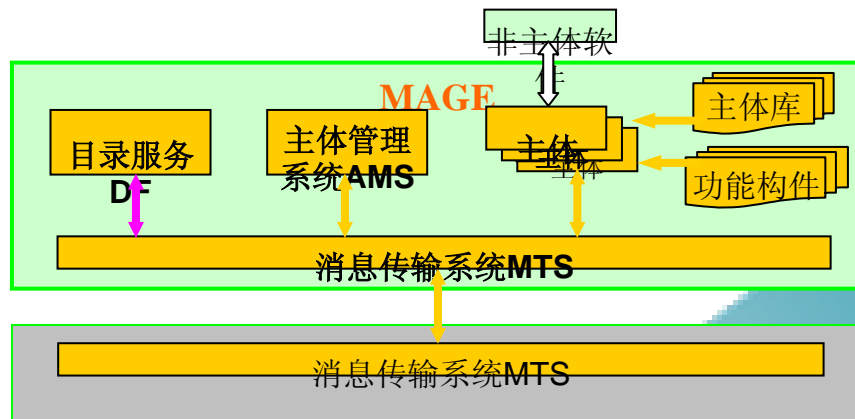
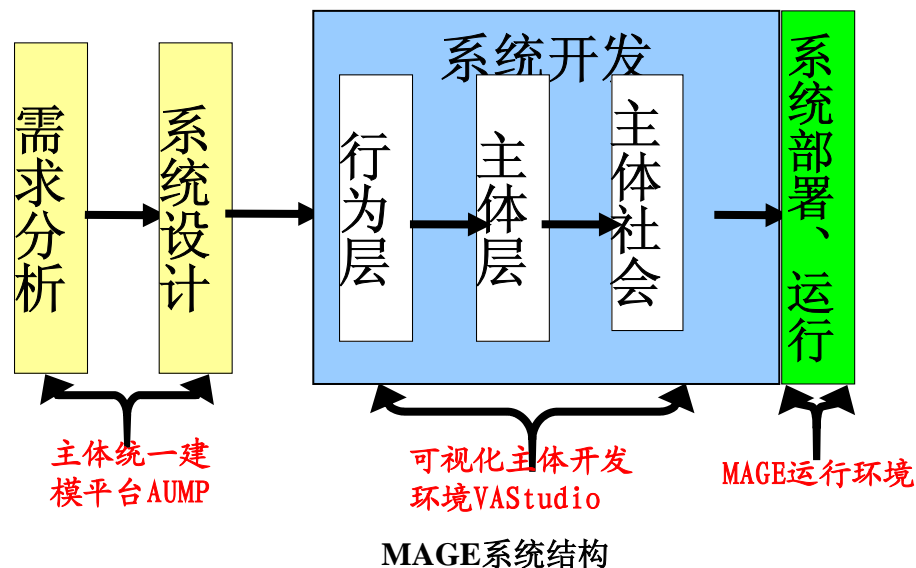
Zhongzhi Shi. Autonomic Semantic Grid. Keynote speaker. IFIP AIAI2005, Sept., 7-9, 2005



多主体环境MAGE

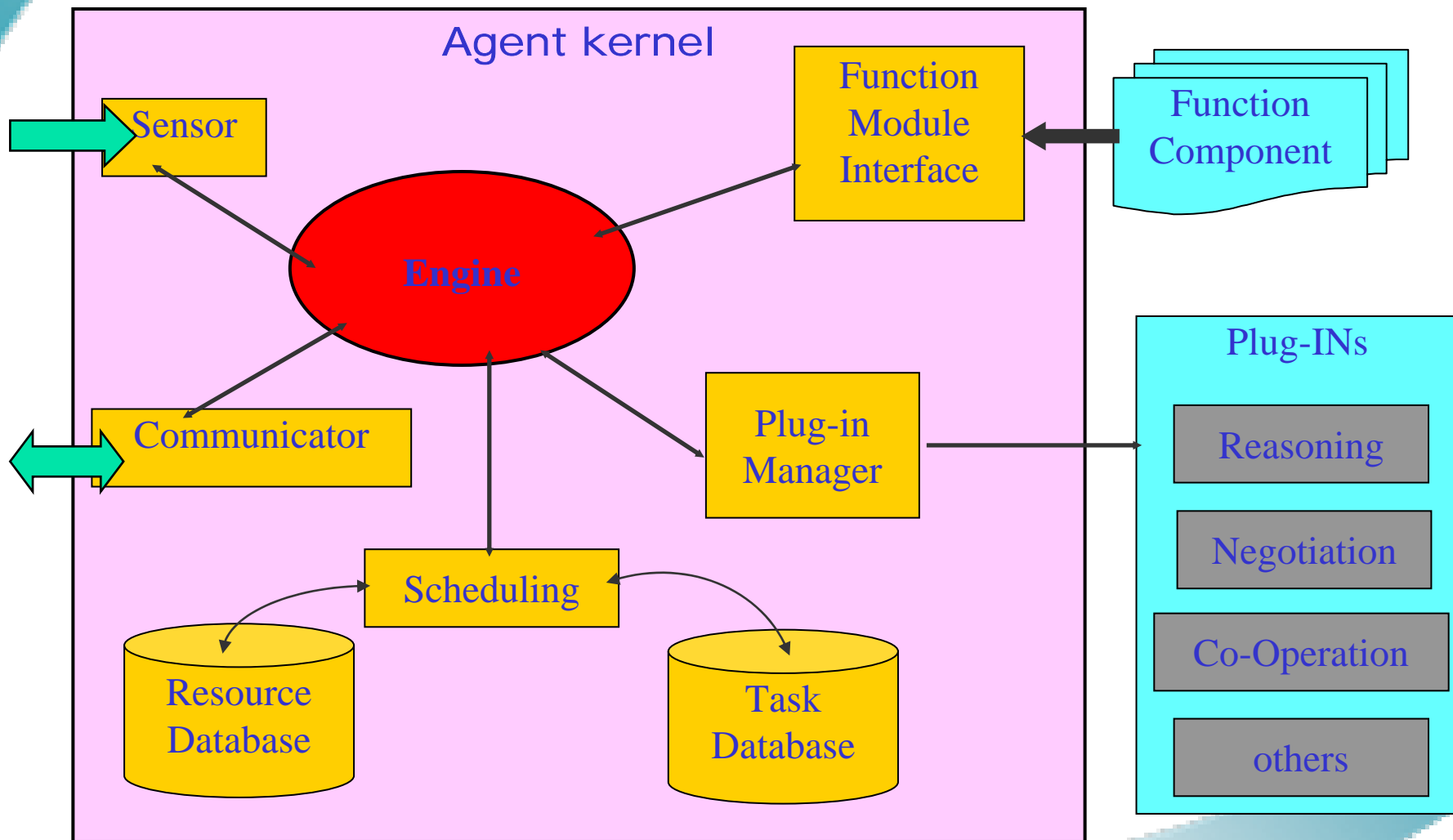
多主体环境MAGE是一种面向主体的软件开发、集成和运行环境，为用户提供需求分析、系统设计、主体生成以及系统运行等多种工具。用户通过构建新的软件或重用已有软件，可以方便地开发各种大型复杂应用。

Zhongzhi Shi. MAGE: Multiagent Environment for Humanized Systems. Invited speaker, ISHS2005





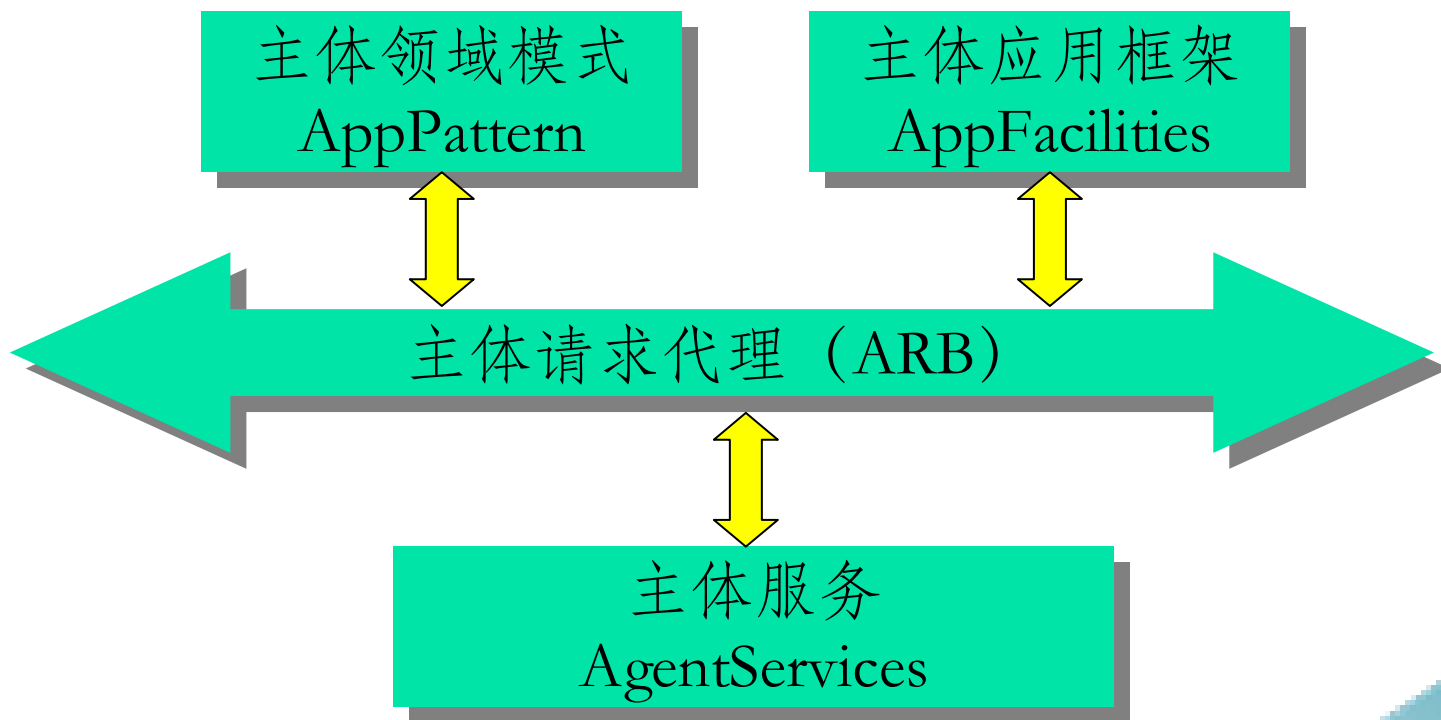
MAGE 主体系统结构





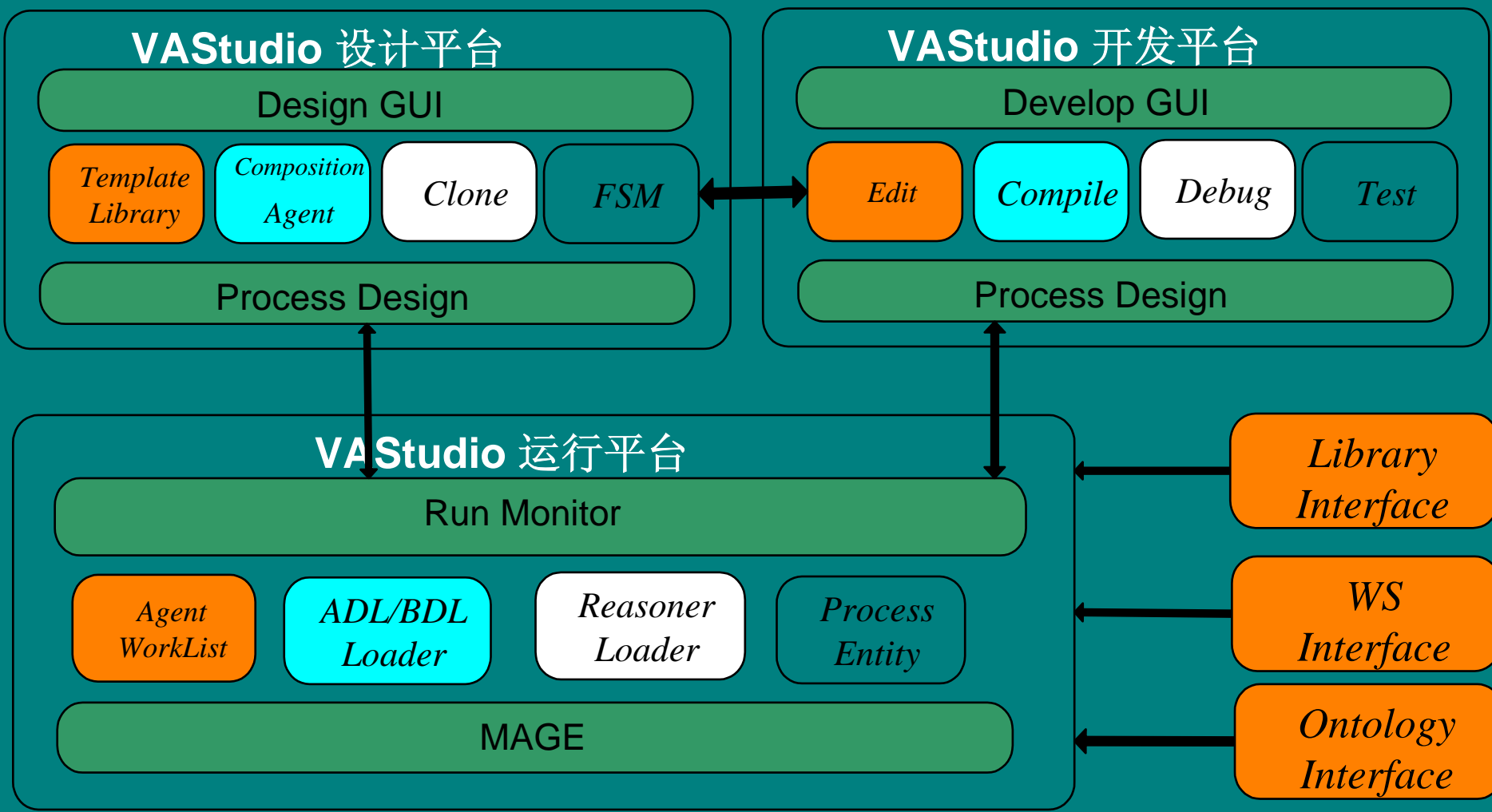
面向主体的软件开发环境

大型软件系统CARBA



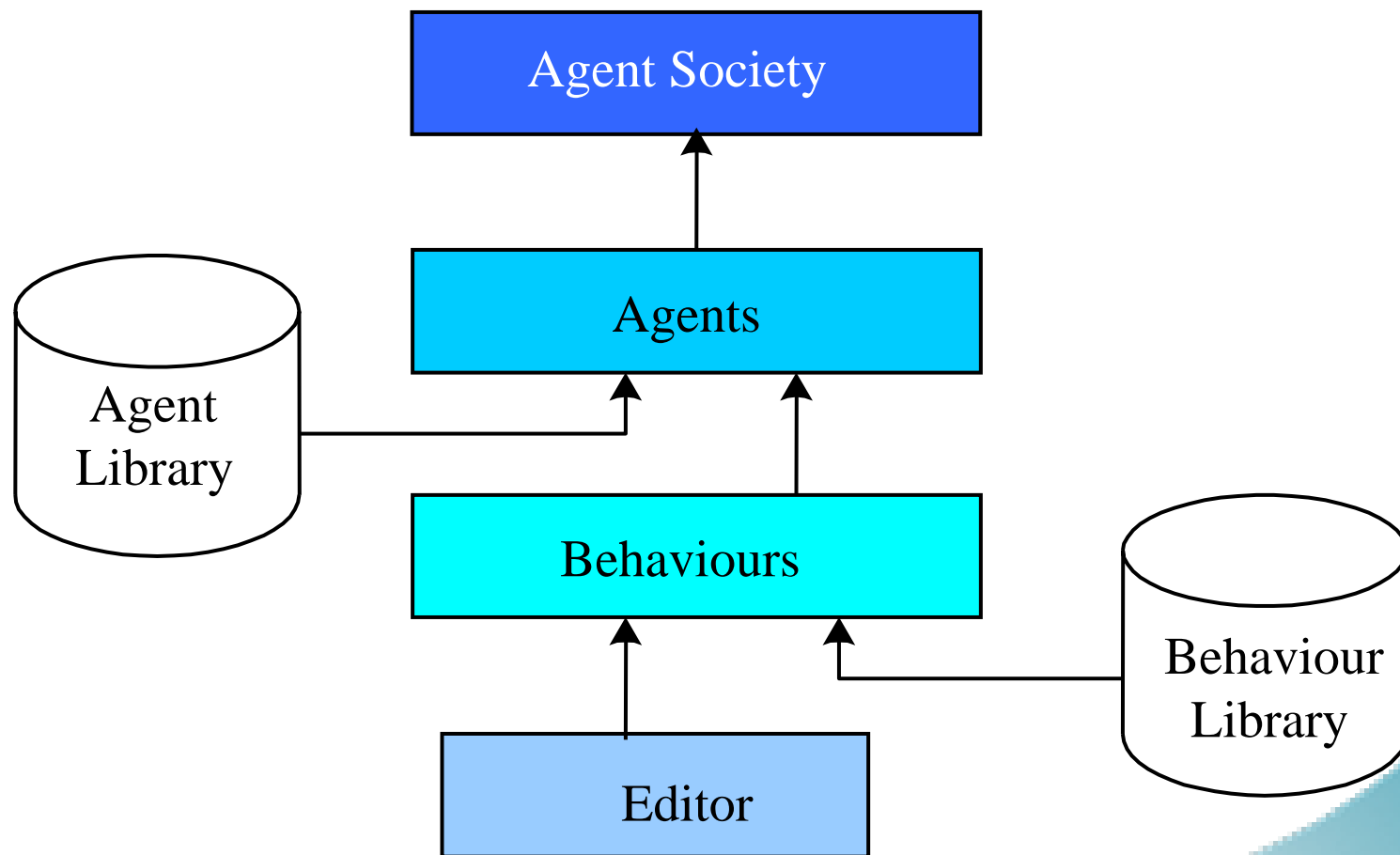


可视化主体开发环境 VASudio





VStudio 系统结构





行为库

- 数据包

```
package intsci.ace.data
```

- 神经网络包

```
package intsci.ace.neural
```

- 学习包

```
package intsci.ace.learning
```

- 数据挖掘包

```
package intsci.ace.mining
```

- 语言处理包

```
package intsci.ace.language
```



行为库

- 图形处理包

`package intsci.ace.graphics`

- 图象处理包

`package intsci.ace.image`

- 搜索引擎包

`package intsci.ace.search`

- 专家系统包

`package intsci.ace.expert`

- 模型包

`package intsci.ace.model`

- 决策支持包

`package intsci.ace.decision`



VASstudio系统界面

The screenshot displays the Visual Agent Studio (VASstudio) Ver3.0.00 interface. The main window is titled "Visual Agent Studio - VASstudio Ver3.0.00" and features a menu bar with options: File, Edit, Search, Insert, View, Behaviour, Agent, Society, Option, Tools, Window, and Help. Below the menu bar is a toolbar with various icons for file operations and development tools.

The interface is divided into several panes:

- Behaviours Pane:** Located on the top left, it shows a tree view of behaviours: DBQueryBehav, DrawCircleBehav, DrawRectangleBehav (selected), and MyBehaviour.
- Behaviour, Agent, Society Tabs:** Located below the Behaviours pane, these tabs allow switching between different development contexts.
- Code Editor:** The central area displays the code for "DrawRectangleBehav.java". The code includes a copyright notice for Key Lab of IIP, CAS, and several import statements for VASstudio and standard Java classes (java.applet.Applet, javax.swing.*, java.awt.*, java.util.*, java.sql.*). A context menu is open over the code, listing actions such as Undo (Ctrl-Z), Redo (Alt-Z), Cut (Ctrl-X), Copy (Ctrl-C), Paste (Ctrl-V), Delete Line (Ctrl-D), Select All (Ctrl-A), Find & Replace (Ctrl-F), Save (Ctrl-S), Compile, and Close (Alt-F4).
- Console Pane:** Located at the bottom, it shows the current directory path: E:\space\com\stu\vastudio\build\class>.
- Bottom Tabs:** Console, Compile, Methods, and Search.

At the bottom of the window, the status bar indicates "Current Window : DrawRectangleBehav" and "Line: 4/177 Column: 24:24 - 2%".

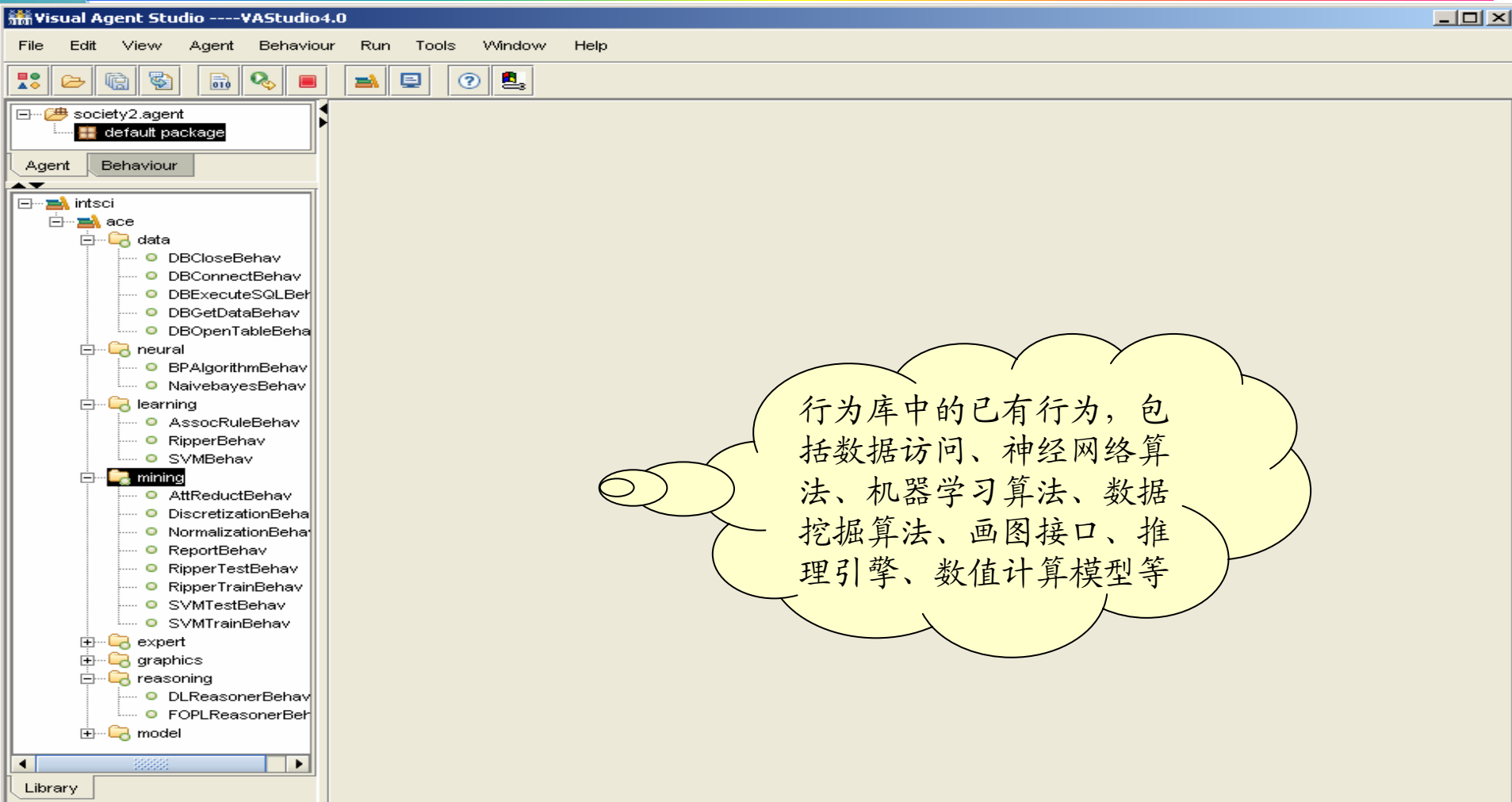


可视化主体开发环境VASTudio





可视化主体开发环境VStudio



@copyright 2005, Mailto: linf@ics.ict.ac.cn, jiaj@ics.ict.ac.cn



用 VASstudio 编写一个主体例 (1)

The screenshot displays the Visual Agent Studio (VASstudio) interface. The main window is titled "Agent1*" and shows a "Behaviour Information" pane with the following text:

This Behaviour aimed at drawing circle.
There are three parameters:
Add template behaviour to the agent
2) parameter-2: Y coordinate of the center of the circle;
3) parameter-3: Radius of the circle.
ACLMessage:
Action: INFROM
Content: "draw a circle"
(Case Sensitive)

To the right, the "Parameters" section is visible, showing a dropdown menu with "(---Please Select Parameter---)", a "Parameters Type" field, and a "Description" field.

A yellow thought bubble contains the text: "1. 加入行为库中的现有行为" (1. Add existing behavior from the behavior library).

The interface also shows a "Behaviour Library" pane on the left with a tree view of various behaviors, and a "Library" pane at the bottom left.

At the bottom of the window, the text "@copyright 2005, Mailto: linf@ics.ict.ac.cn,jiayj@ics.ict.ac.cn" is visible.



用 VASstudio 编写一个主体例 (2)

Visual Agent Studio --- VASstudio 4.0

File Edit View Agent Behaviour Run Tools Window Help

Set Behaviour Parameters

Behaviour Information

3) parameter-3: Radius of the circle.

ACLMessage:
Action: INFROM
Content: "draw a circle"
(Case Sensitive)

1) The first parameter is X coordinate of the circle: 60
2) The second parameter is Y coordinate of the circle: 80
3) The third parameter is radius (int): 50

OK Cancel

2. 填写行为参数

intsci

ace

data

- DBCloseBehav
- DBConnectBehav
- DBExecuteSQLBeh
- DBGetDataBehav
- DBOpenTableBeha

neural

- BPAlgorithmBehav
- NaivebayesBehav

learning

- AssocRuleBehav
- RipperBehav
- SVMBehav

mining

- AttReductBehav
- DiscretizationBeha
- NormalizationBeha
- ReportBehav
- RipperTestBehav
- RipperTrainBehav
- SVMTestBehav

Agent1

Agent1

Parameters

(---Please Select Parameter---)

Parameters Type

parameters Description

all the behaviours that have been added to this agent

intsci.ace.graphics.DrawCircleBehav

Design Source

@copyright 2005, Mailto: linf@ics.ict.ac.cn,jiayj@ics.ict.ac.cn

用VASstudio编写一个主体例



(3)

```
55  * execution model is still sequential and no behaviour scheduling.
56  * is active yet..
57  *.
58  * This method can be used for ordinary startup tasks such as.
59  * <b>DF</b> registration, but is essential to add at least a.
60  * <code>Behaviour</code> object to the agent, in order for it to be.
61  * able to do anything..
62  * @see mage.core.Agent#addBehaviour (Behaviour b).
63  * @see mage.core.behaviours.Behaviour.
64  */.
65  protected void setup () { //VASTUDIOAGENTBOOKMARK: setup.
66
67      try{
68          Class c = Class.forName("intsci.ace.graphics.DrawCircleBehav");.
69          Class tmpAgent = Class.forName("mage.core.Agent");.
70          Class[] para = {tmpAgent};.
71          java.lang.reflect.Constructor con = c.getConstructor(para);.
72          mage.core.Agent[] as = {this};.
73          vastudio.spec.InterfaceSpec is = (vastudio.spec.InterfaceSpec)con.newInstance(as);.
74          is.setParameter(0, ""+"60");.
75          is.setParameter(1, ""+"80");.
76          is.setParameter(2, ""+"50");.
77          this.addBehaviour((Behaviour)is);.
78      }.
79      catch (Exception e){
80          e.printStackTrace();.
81      }.

```

3. 主体编辑完成，
代码已自动生成

@copyright 2005, Mailto: linf@ics.ict.ac.cn,jiayj@ics.ict.ac.cn



竞拍系统

1. 编写拍卖发起主体
(即拍卖方): 采用
有限状态自动机方
式, 画出状态图转换
并指定通信方式即
可, 代码将自动生成



竞拍系统

2. 编写两个拍卖响应主体（即竞拍方）：同样采用有限状态机方式，两个同样的竞拍主体可以直接复制得到

@copyright 2005, Mailto: linf@ics.ict.ac.cn jiayj@ics.ict.ac.cn



竞拍系统

3. 拍卖系统开发完成，编译通过



竞拍系统

4. 在MAGE
运行环境中
运行系统



竞拍系统

RMA@linfen:5689/MAGE - MAGE Remote Agent Management GUI

File Actions Tools Platforms Help

MAGE

AgentPlatforms

- "linfen:5689/MAGE"
 - Main-Container
 - auction.agent.In
 - df@linfen:5689/
 - RMA@linfen:5689/
 - auction.agent.R**
 - ams@linfen:5689/
 - auction.agent.R

Name: Class:

Language: Ontologies:

Current State

- Active
- Suspen...
- Wait...

Sent/Received messages Pending messages

*	Time	Performative	Sender/Receiver	Content
	2006-03-10 1...	REQUEST	ams@linfen:...	((action (age...
	2006-03-10 1...	AGREE	ams@linfen:...	((action (age...
	2006-03-10 1...	INFORM	ams@linfen:...	((done (actio...
	2006-03-10 1...	INFORM	auction.agent...	inform-start...
	2006-03-10 1...	CFP	auction.agent...	2000
	2006-03-10 1...	PROPOSE	auction.agent...	2789
	2006-03-10 1...	ACCEPT-PR...	auction.agent...	accept-propo...

运行结果
(1):
一个竞拍主体的
运行过程



竞拍系统

RMA@linfen:5689/MAGE - MAGE Remote Agent Management GUI

File Actions Tools Platforms Help

AgentPlatforms

- "linfen:5689/MAG
 - Main-Containe
 - auction.ag
 - df@linfen:
 - RMA@linfe
 - auction.ag
 - ams@linfe
 - auction.ag

Name: auction.agent.InitAgent Class: auction.agent.InitAgent

Language: Ontologies:

Current state: Active

Sent/Received messages

*	Time	Performative	Sender/Receiver	Content
	200...	REQUEST	ams@linfen:5689/MAGE;	((action...
	200...	AGREE	ams@linfen:5689/MAGE	((action...
	200...	INFORM	ams@linfen:5689/MAGE	((done (...
	200...	INFORM	auction.agent.ResAgent1...	inform-...
	200...	CFP	auction.agent.ResAgent1...	2000
	200...	PROPOSE	auction.agent.ResAgent2...	2789
	200...	PROPOSE	auction.agent.ResAgent1...	2533
	200...	ACCEPT-PRO...	auction.agent.ResAgent2...	accept-...
	200...	REJECT-PROP...	auction.agent.ResAgent1...	reject-p...

Pending messages

运行结果
(2):
拍卖主体的运
行过程



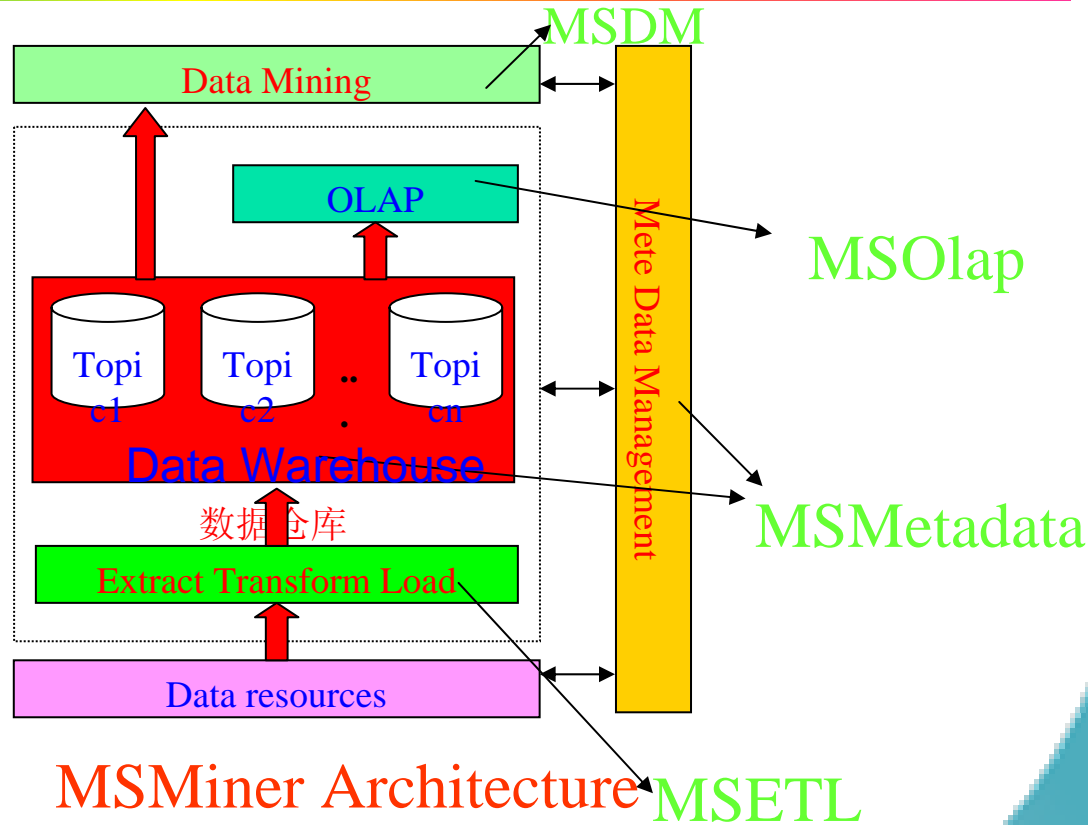
多主体系统比较

		AgentBuilder	Jack	Zeus	MAGE
分析 Analysis	Completeness:	★★★	★★★	★★★★	★★★★★
	Applicability:	★★★	★★★	★★★	★★★★
	Complexity:	★★★★★	★★★	★★★★★	★★★★
	Reusability	★★★	★★★	★★★	★★★★★
设计Design	Completeness:	★★★	★	★★★★	★★★★★
	Applicability:	★★★	★	★★★	★★★★
	Complexity:	★★★★	★	★★★	★★
	Reusability	★★	★★	★★★	★★★★
开发 Development	Completeness:	★★★★★	★★★★	★★★★★	★★★★★
	Applicability:	★★★	★★★★★	★★★	★★★★
	Complexity:	★★★★	★★	★★★★	★★★
	Reusability	★★	★★★★★	★★	★★★
运行 Deployment	Completeness:	★★★	★★★	★★★★	★★★★★
	Applicability:	★★★	★★★★	★★★	★★★
	Complexity:	★★★★	★★★	★★★★	★★★★★
	Reusability				★



数据挖掘平台 MSMiner

多策略知识挖掘软件 MSMiner 为企业决策和智能信息处理提供数据挖掘总体解决方案。系统采用功能强大的元数据作为调度中心，实现了数据仓库与数据抽取、转换、装载 (ETL)、数据挖掘、联机分析处理 (OLAP) 的有机集成和各种数据挖掘算法的无缝连接。系统提供了多种数据挖掘算法。

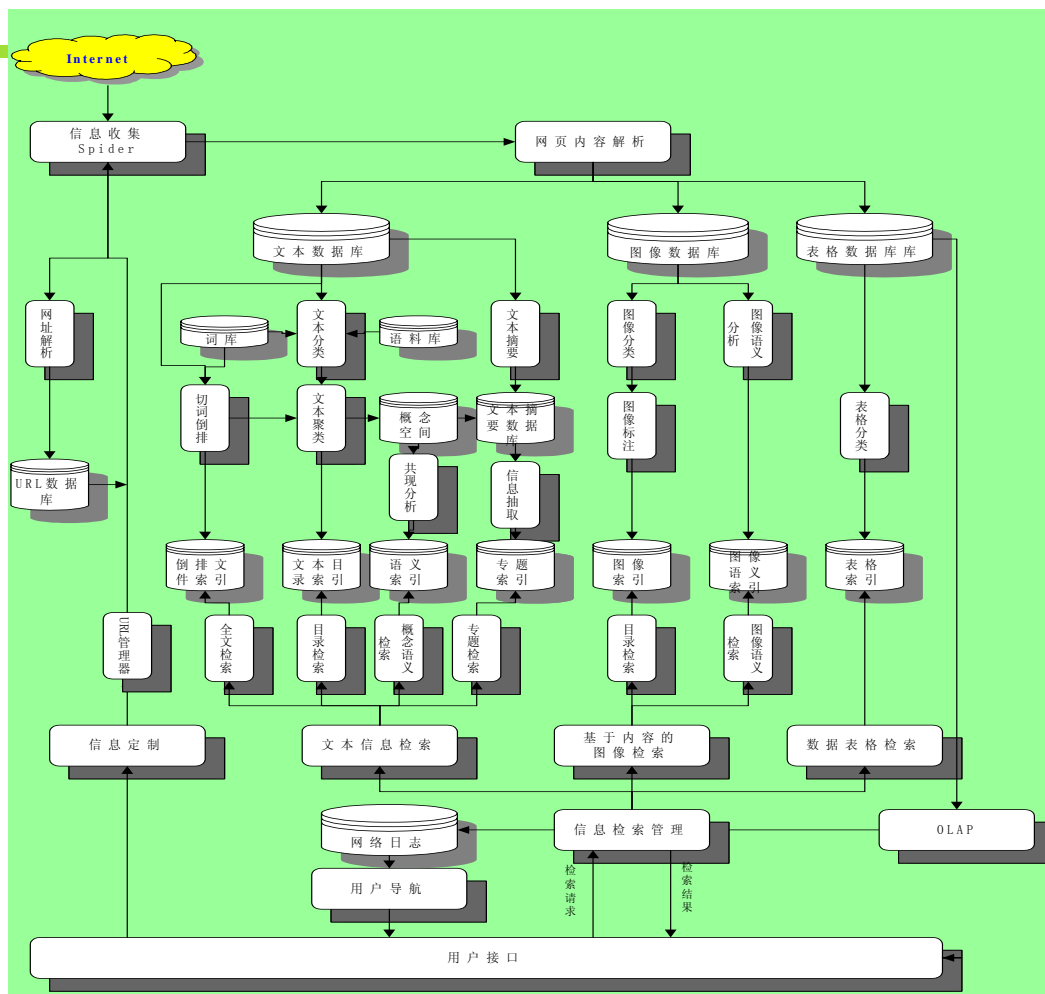


Zhongzhi Shi, Youping Huang, Qing He, Lida Xu, Shaohui Liu, Liangxi Qin, Ziyang Jia, Jiayou Li. MSMiner-A Developing Platform for OLAP. Decision Support Systems.



智能搜索引擎 GHunt

智能搜索引擎GHunt是网络信息智能获取与处理系统，支持分布式的网络信息并行搜索与内容过滤；采用文本挖掘自动建立概念语义空间和事件来龙去脉，提供高效的基于语义的文本信息检索、基于内容的图像检索，以及个性化的专题信息推送服务。

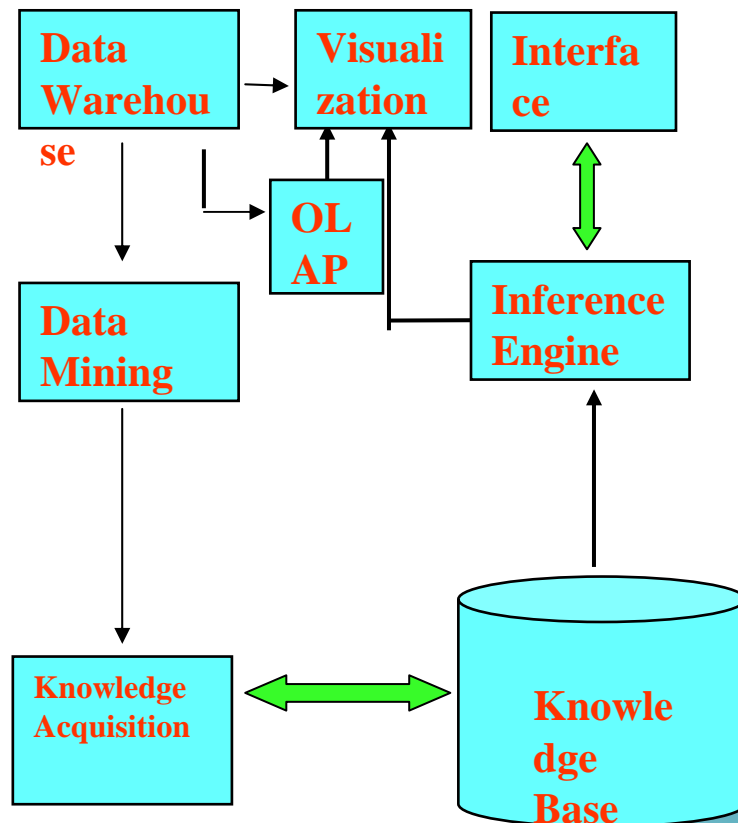


Zhongzhi Shi, Qing He, Ziyang Jia and Jiayou Li. Intelligence Chinese Document Semantic Indexing System. International Journal of Information Technology and Decision Making, Vol.2, No.3, 2003: 407-424.
2006-8-21



专家系统工具OKPS

OKPS系统采用面向对象的知识表示方法来表示知识，并通过通用的推理机制根据用户的需求进行推理。专用的推理控制语言保证了该专家系统工具可以构造功能足够强大和灵活的专家系统。



Zhongzhi Shi, Ping Luo, Yalei Hao, Guohe Li, Markus Stumptner, Qing He and Gerald Quirchmayr. INTELLIGENT TECHNOLOGY FOR WELL LOGGING ANALYSIS. IIP2004, 373-382



知识产权

软件著作权：5项

MAGE



MSMiner



MIRES



OKPS



GHunt



成果评价

主体网格智能平台AGrIP

- **鉴定意见：**2006年1月11日由中国科学院主持召开了“面向智能主体的软件工程研究——主体网格智能平台**AGrIP**”的科技成果鉴定会。鉴定委员会一致认为：本项目取得的动态描述逻辑**DDL**、面向智能主体的软件开发方法、主体网格软件架构和服务模式、基于本体的知识管理等研究成果达到国际先进水平；**研制的主体网格智能平台AGrIP在功能集成方面位于国际上同类系统的前列，具有重大的应用前景。**

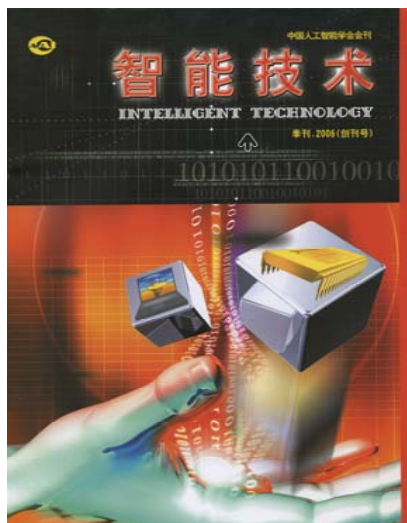




成果评价

在国际会议SELMAS 2006上应邀作主题报告:

**Agent Grid Intelligence
Platform for Collaborative
Working Environment**



2006-8-21

智能技术发刊词

中国人在人工智能研究方面已经取得了一系列令人鼓舞的成果：创造了包括几何定理证明的吴氏方法、...**主体网格智能理论**等中国的**AI**品牌，表现了旺盛的创新力。

主体计算-史忠植



内容提要

- 概述
- 智能主体的认知模型
- 基于主体的SOA
- 主体协同工作
- 主体网格智能平台AGrIP
- 应用
- 结束语



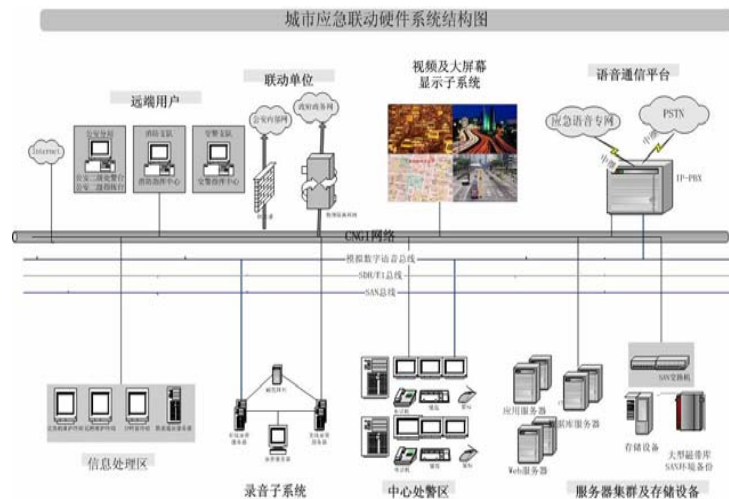
推广应用

主体网格智能平台**AGrIP**已在国家安全、能源、服务业、教育、国防等领域**40**多个单位进行了推广应用：

北京曙光天演信息技术有限公司、中国科学院计算技术研究所等联合采用主体网格智能平台**AGrIP**研制《城市应急联动与社会综合服务系统》**GEIS**。



2006-8-21



- 省会城市**30**多个 **2000**万元/每个城市
 - 地级城市**300**多个 **1000**万元/每个城市
 - 县级城市**2800**多个 **500**万元 /每个城市
- 全国合计总建设费用约**100**亿元，其中软件系统费用约**40**亿元。



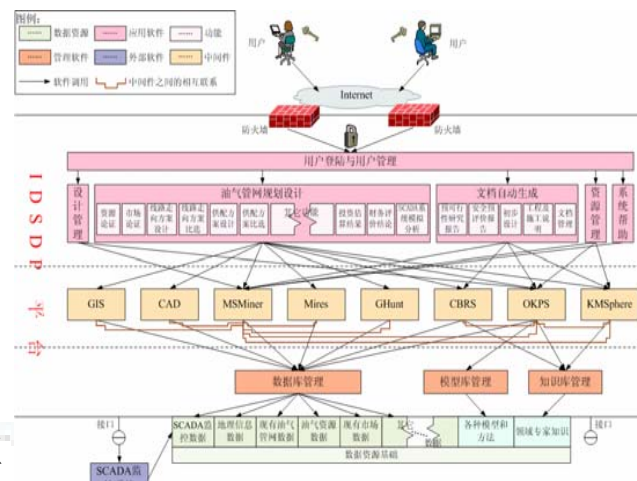
推广应用

国家重大项目提供关键技术和支持平台



国家电网公司科学技术项目“电力供需实验室的开发与建设”，投资1700万

中国石油规划总院“国内油气管网规划决策支持系统” 投资1000万





推广应用

北京普天和平通信技术公司采用AGrIP开发的广告监测系统已在多个省的工商局广告监测中心推广应用，该系统促进了广告监测的客观、及时、高效、准确和自动化，对建立和谐社会作出了贡献

公司利用该项目的技术，新增利税800万元

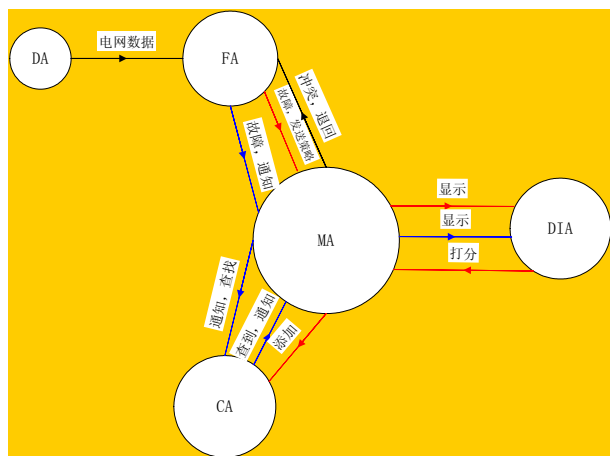


查询片段	平均准确率(%)	平均召回率(%)
广告片段	100	100
新闻片段	66.7	57.1
电视台标识片段	90	90
体育片段	100	100
平均	89.2	86.8



推广应用

日本京都大学、歌山大学、**NTT**等单位合作的《亚洲宽带——异文化协作环境》中，利用主体网络智能平台**AGrIP**为底层支撑平台，构建多文化协作平台。



中国电力科学院采用**AGrIP**开发了国内第一个大规模复杂配电网快速故障恢复决策支持系统, 解决了应用传统方法对大规模网格状多馈线供电配电网进行故障恢复耗时长的的问题





推动科技进步的作用

为国家重大基础研究提供关键技术和支持:

1. 国家**973** “信息技术中的应用理论和高性能软件”项目,中国电力科学院采用**AGrIP**开发了国内第一个大规模复杂配电网快速故障恢复决策支持系统;
2. 国家**973** “中国陆地生态系统碳循环及其驱动机制研究”采用**AGrIP**开发国家陆地生态系统碳收支动态评估和协同决策支持系统;
3. 国家**973**项目“**语义网格的基础理论、模型与方法研究**”中采用**AGrIP**为重要平台之一。



结束语

- 主体模型的探讨
- 主体网格智能理论的研究
- 多主体环境MAGE的开发
- 主体网格智能平台AGrIP为构建大型复杂系统提供协同工作环境



谢 谢

欢迎各位专家、
领导批评指正！



Question!