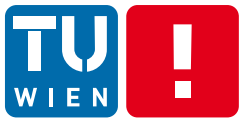


Context-aware Programming for Hybrid and Diversity-aware Collective Adaptive Systems

Hong-Linh Truong and Schahram Dustdar

Distributed Systems Group
Vienna University of Technology
truong@dsg.tuwien.ac.at

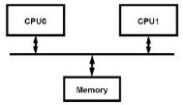
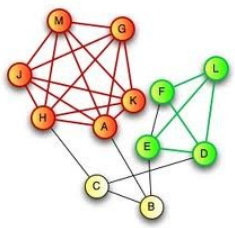
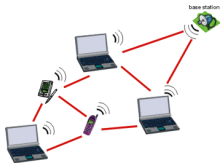
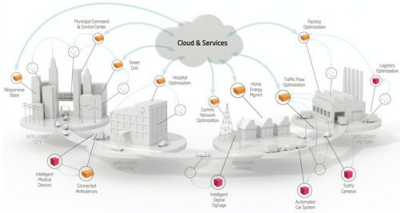
http://dsg.tuwien.ac.at/research/viecom/

Outline

- Background
- Motivation
- Context associated with HDA-CAS
- h^2CAS – using hybrid computing units for HDA-CAS
- Context-aware programming features
- Conclusions and future work

Background

S. Dustdar, H. Truong, "Virtualizing Software and Humans for Elastic Processes in Multiple Clouds – a Service Management Perspective", in *International Journal of Next Generation Computing*, 2012

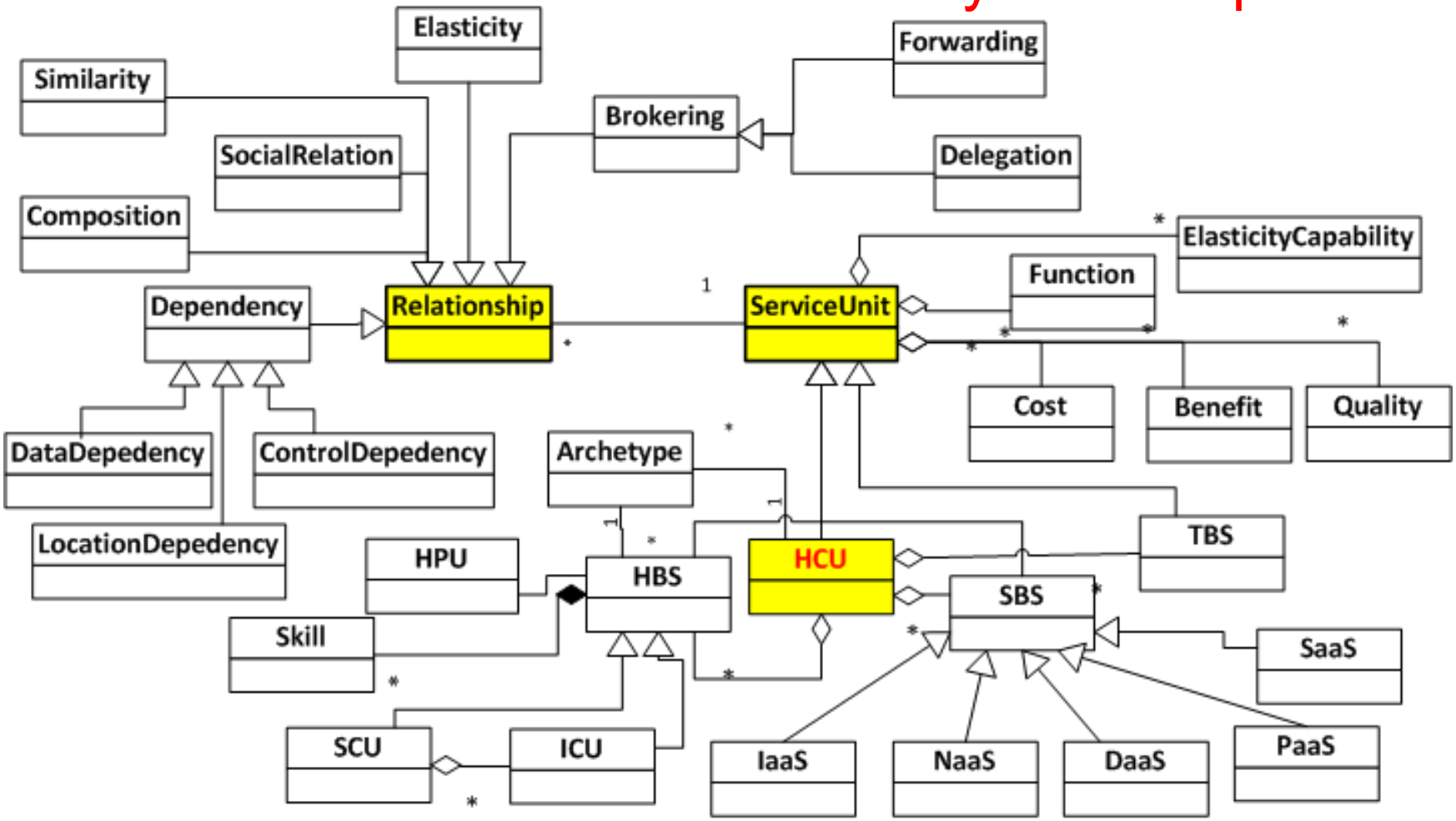
	Machine-based Computing	Human-based Computing	Things-based computing
Processing Unit	  		 
Architecture	 SMP   Grid	  	Ad hoc networks  Web of things  
Comm.	 	  	  

Background (cont.)

- Today's complex problems need hybridity and diversity-aware "collective adaptive systems" (HDA-CAS)
- HDA-CAS
  - Mixture of different types of resources working in concert in the same collective
  - Mixture of different roles performed in the same collective
  - Mixture of different quality from a single collective
  - Mixture of cost/benefit models

Background (cont.)

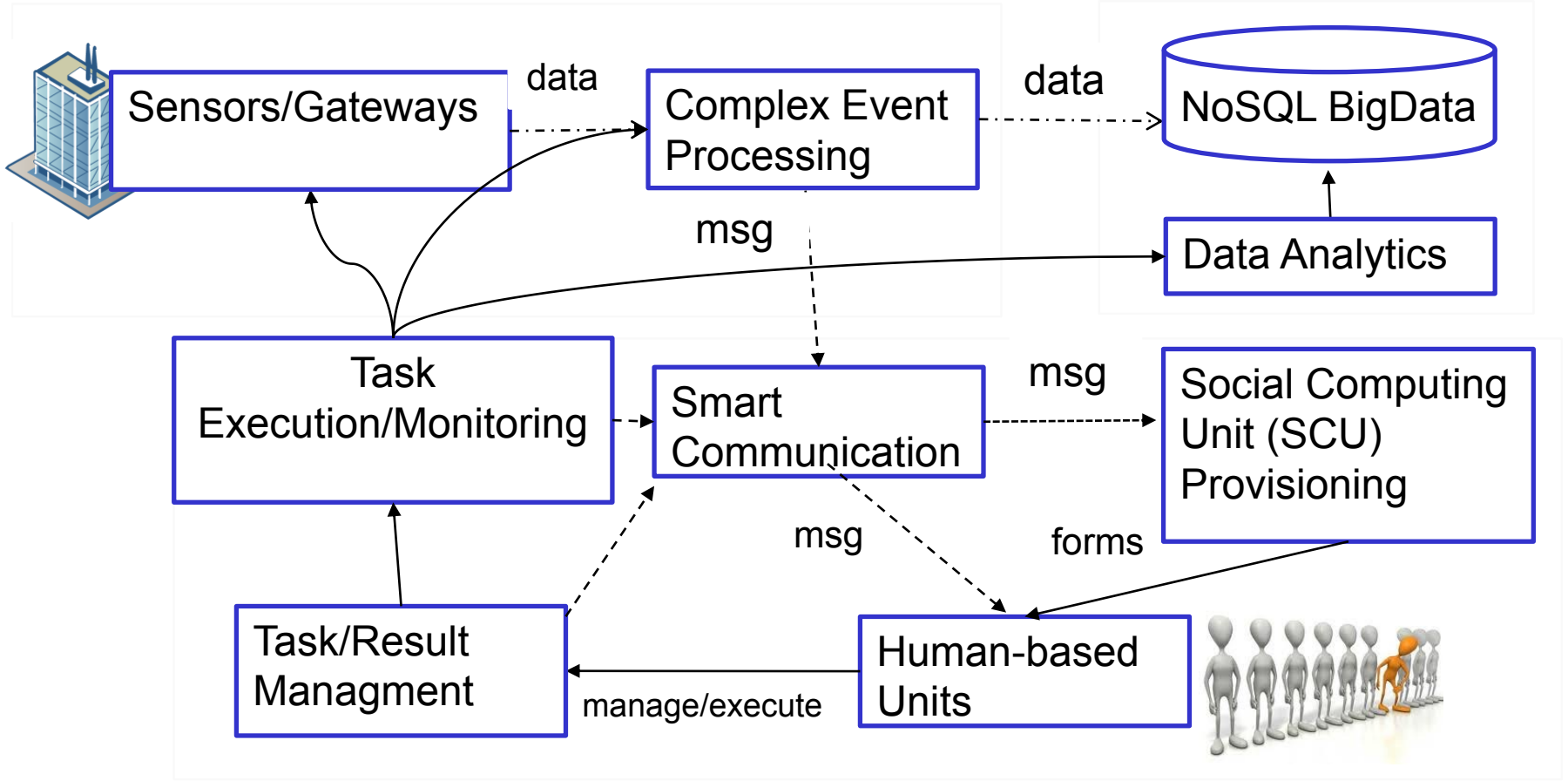
Hybrid compute units



Hong Linh Truong, Hoa Khanh Dam, Aditya Ghose, Schahram Dustdar: Augmenting Complex Problem Solving with Hybrid Compute Units. ICSOC Workshops 2013: 95-110

Motivation

- Cloud-based predictive maintenance – an application of HDA-CAS in the context of BPM



Motivation (cont.)

- HDA-CAS –based solutions
  - Human-, software- and thing-based services for computational analytics, data gathering, and network functions
- Challenges
  - Which are possible contexts during the operation of HDA-CASs? How they affect provisioning mechanisms
  - What are main building blocks for HDA-CAS?
  - Which are main programming features to support these operational contexts?

This talk focuses on conceptual view, requirement and analysis

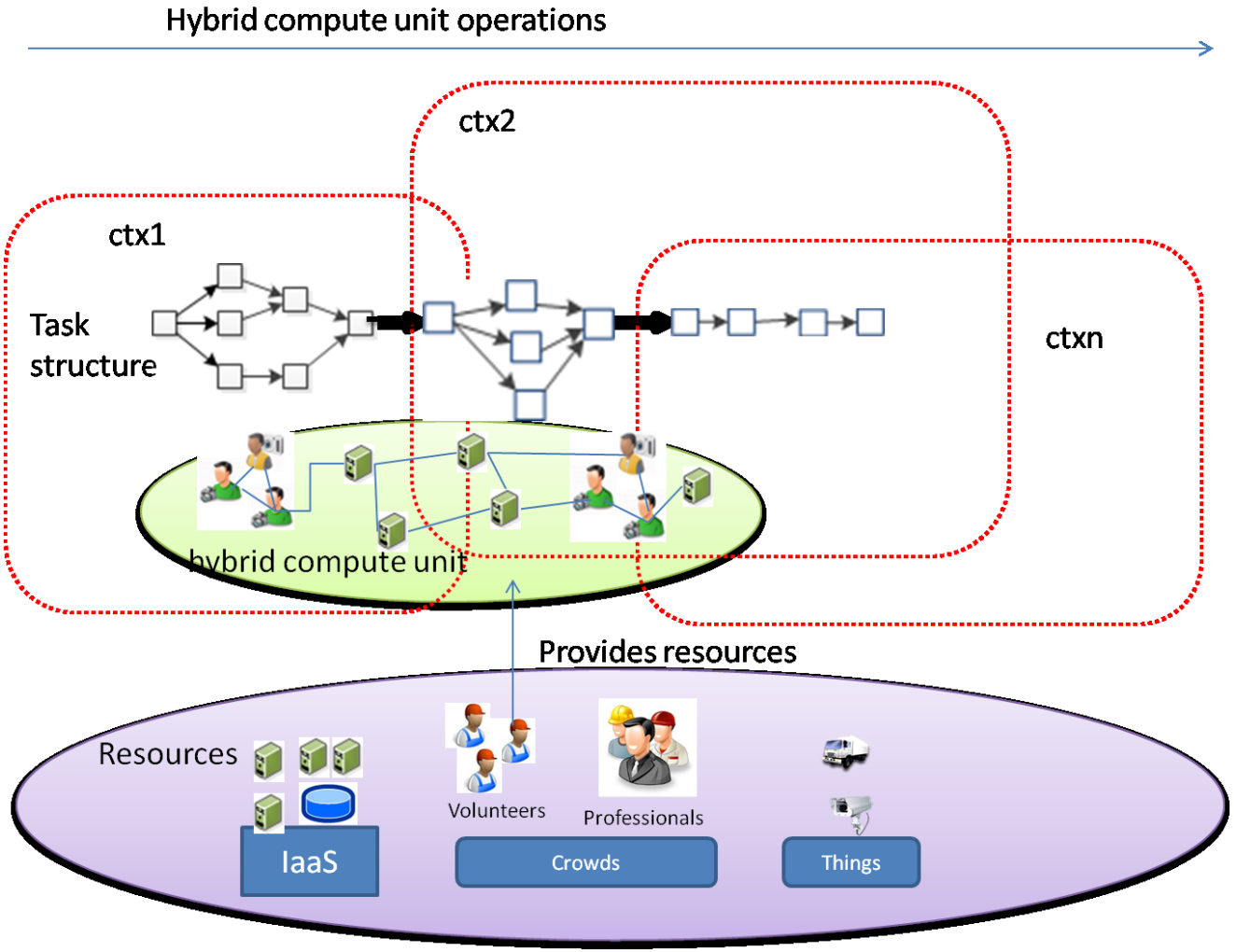
Context-aware programming HDA-CASs – Goals

- Understanding context in order to develop
  - Right constructs for specifying what constitutes a HDA-CAS
  - Tools and middleware for deploying, provisioning and instantiating HDA-CASs based on their specifications, and
  - Means for programming the control and reconfiguration of HDA-CASs at run-time.

Context of HDA-CAS

- What: tasks and quality of results, and structures of the HDA-CAS
- Who/Which: constituting units for computation/data/network functions as well as for monitoring/control/management functions
- How: coordination and elasticity mechanisms that control the operation of the HDA-CAS
- When: a determined time frame for the above-mentioned specific What/Who/Which/How.

Context and HDA-CAS



Hybridity and elasticity in context

- Hybridity/Diversity
  - Hybrid processing units for computation/data/network functions, hybrid architectures, and hybrid communication protocols
- Elasticity
  - Processing units: add/remove/replace processing units
  - Architecture: provision different static and runtime topologies for different types of units, and change different protocols/algorithms within monitoring/control/management units.
  - Communications: add/remove/replace/reconfigure communication protocols

h²CAS - HCU-based HDA-CAS

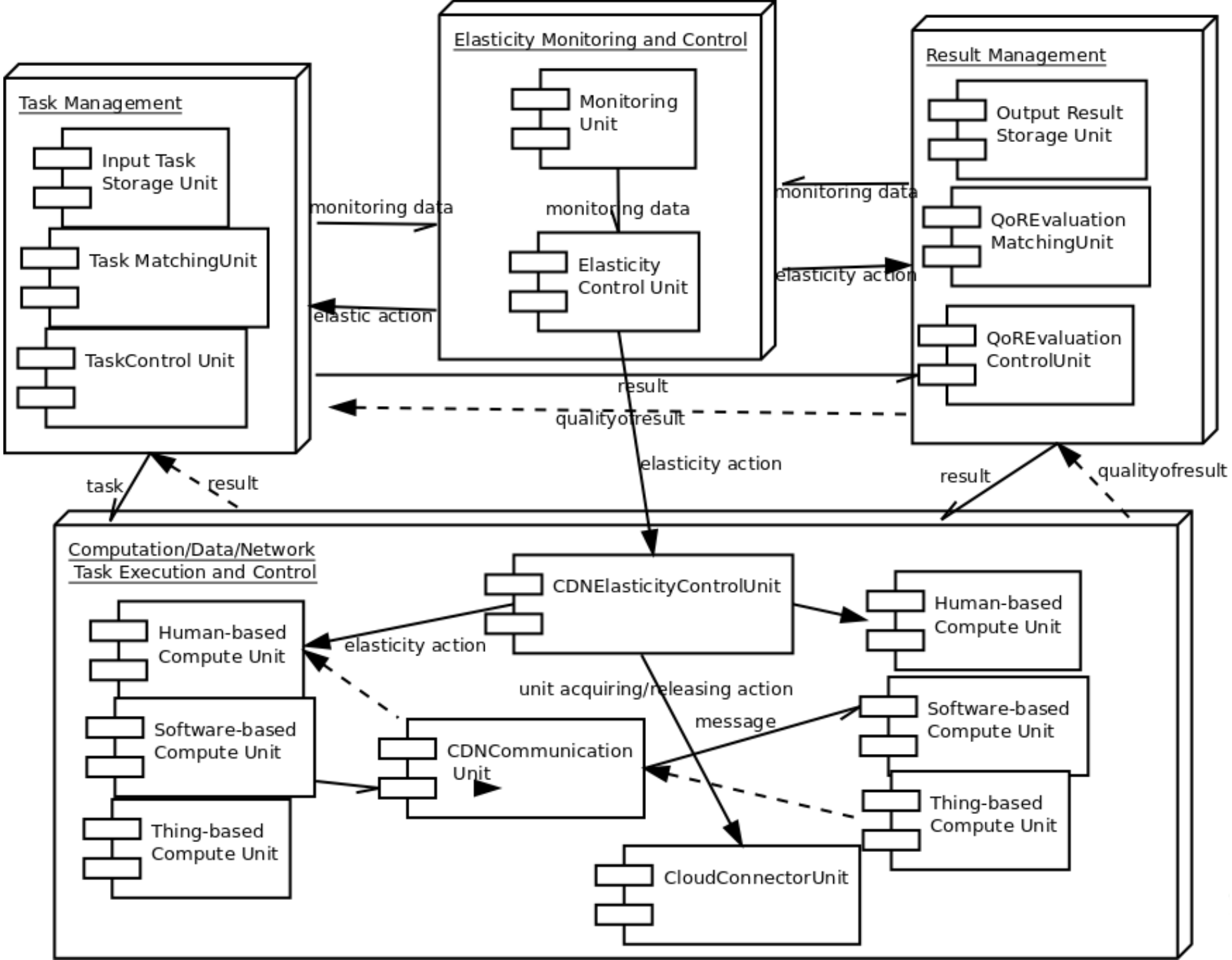
How to program HDA-CAS?

Our approach: using hybrid compute units (HCU) provisioned in the cloud

A HCU-based HDA-CAS (h²CAS) includes a set of service units which can be *software-based services, human-based services and thing-based services* that can be provisioned, *deployed and utilized as a collective on-demand* based on different quality, pricing and incentive models.

h²CAS- Main building blocks

- Task Management
- Result Management
- Computation/Data/Network Task Execution and Control
- Elasticity Monitoring and Control


Task Management
Input Task Storage Unit
Task MatchingUnit
TaskControl Unit
Elasticity Monitoring and Control
Monitoring Unit
Elasticity Control Unit
Result Management
Output Result Storage Unit
QoREvaluation MatchingUnit
QoREvaluation ControlUnit
monitoring data
monitoring data
monitoring data
elastic action
elasticity action
result
qualityofresult
task
result
elasticity action
result
qualityofresult
Computation/Data/Network Task Execution and Control
CDNElasticityControlUnit
Human-based Compute Unit
Software-based Compute Unit
Thing-based Compute Unit
elasticity action
unit acquiring/releasing action
message
CDNCommunication Unit
Human-based Compute Unit
Software-based Compute Unit
Thing-based Compute Unit
CloudConnectorUnit

Context-aware Programming Features

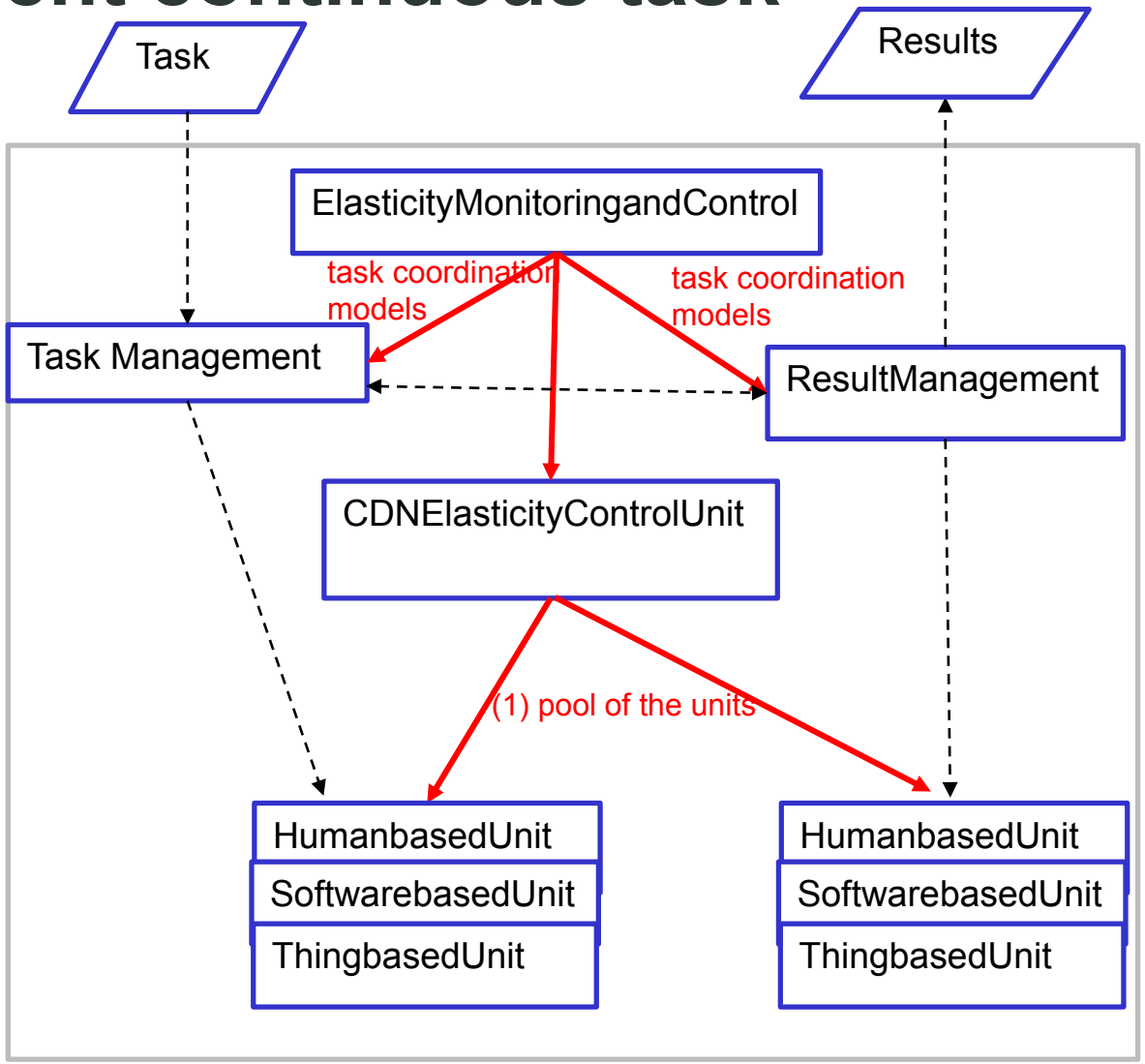
- Analysis of some main provisioning and operational contexts
  - consumer-generated independent continuous task
  - consumer-generated dependent task
  - evolving independent task
  - evolving dependent task

To develop suitable techniques for HDA-CAS provisioning, e.g.:

- Enable context switching in complex problem solving
- Achieve quality and cost/benefit elasticity by utilizing resource elasticity

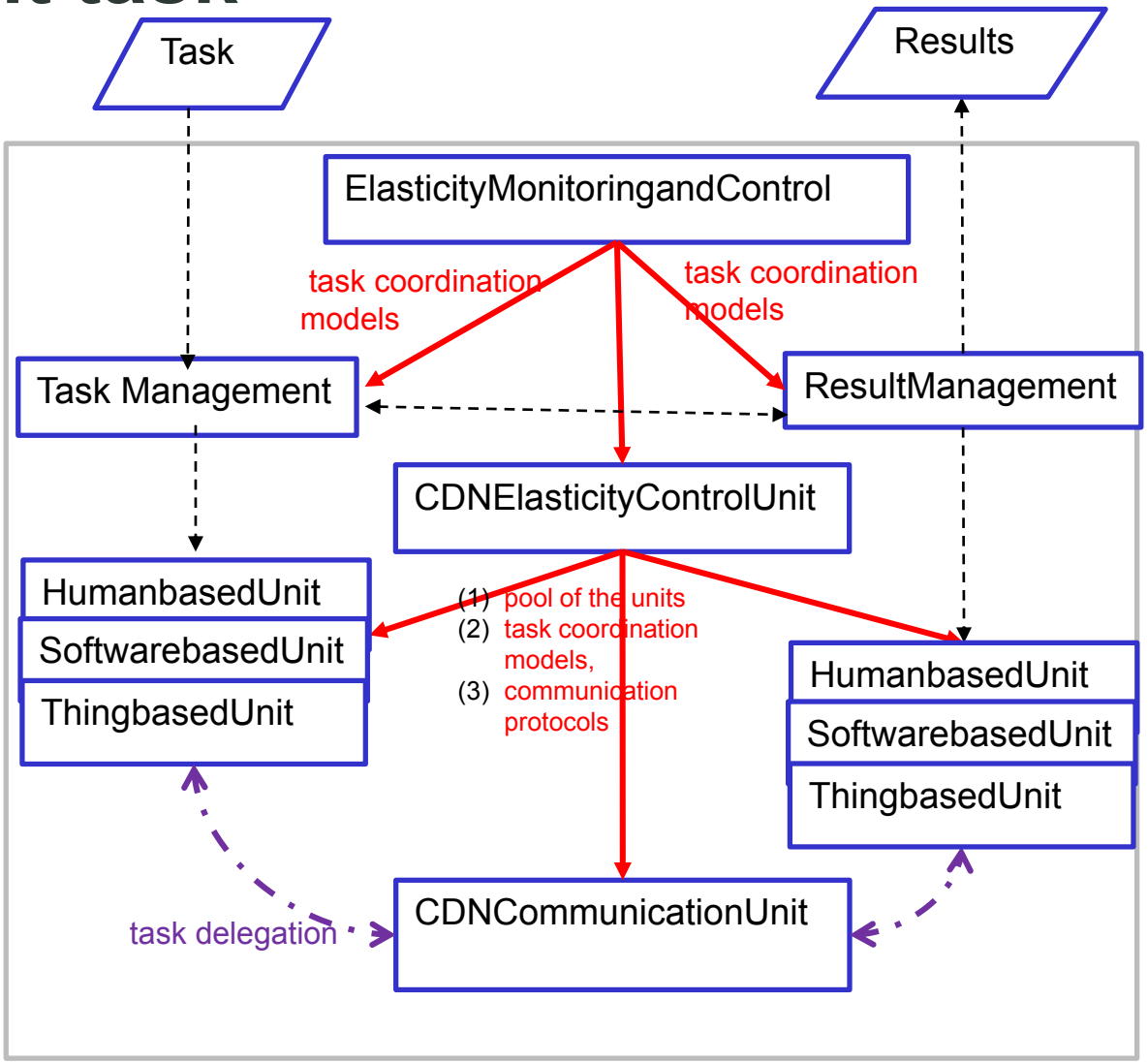
Ctx1: consumer-generated independent continuous task

- Continuous flow of independent tasks (batch jobs, crowdsourcing style)
- One level of task coordination (TaskManagement/Result Management)



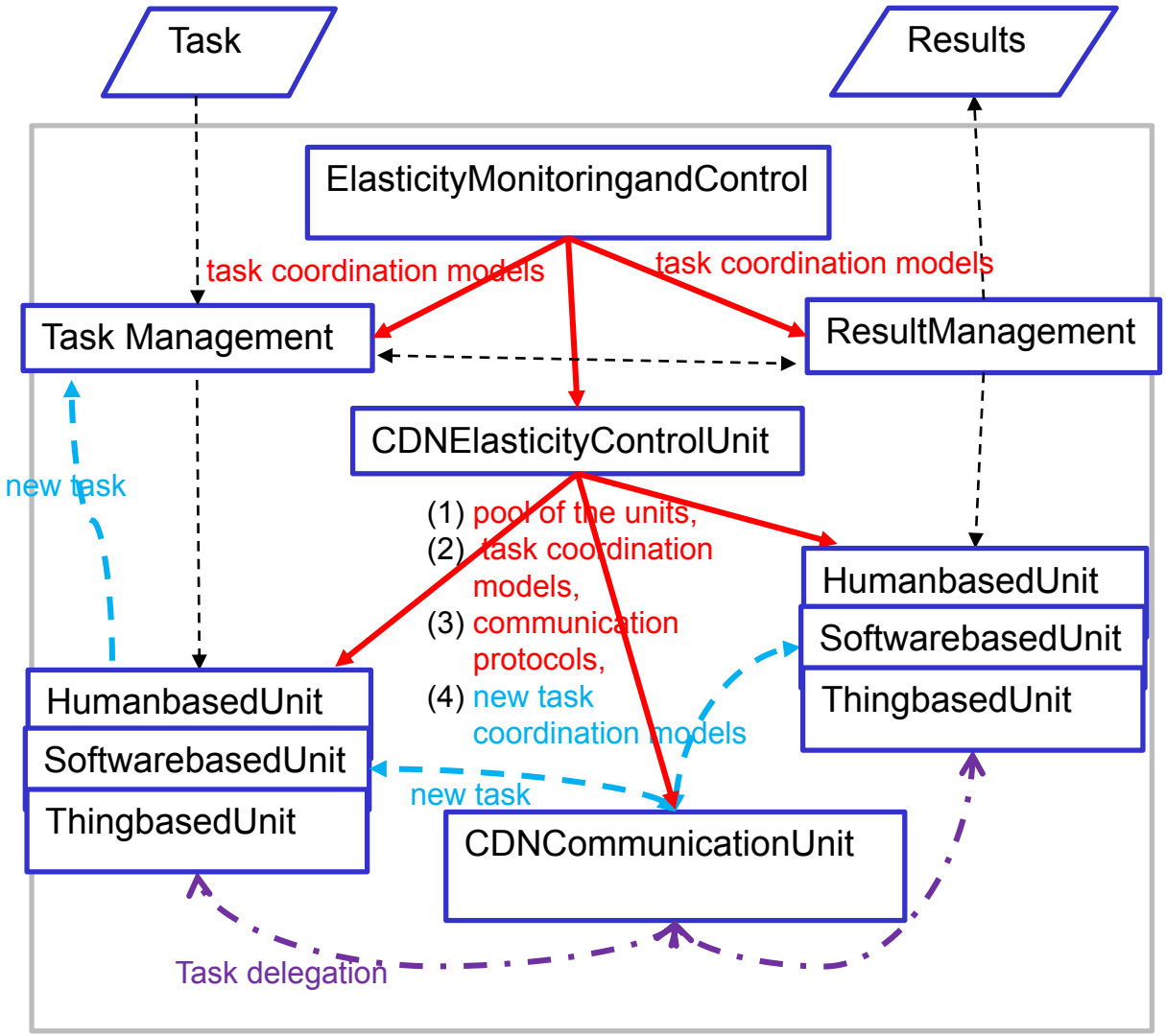
Ctx2: consumer-generated dependent task

- Tasks are dependent, but complete (workflow style)
- Require coordination and unit formation models for dependent tasks
- Single level of task coordination or multi-level task coordination
- Interactive elasticity controls
- Communications needed for task delegation among units



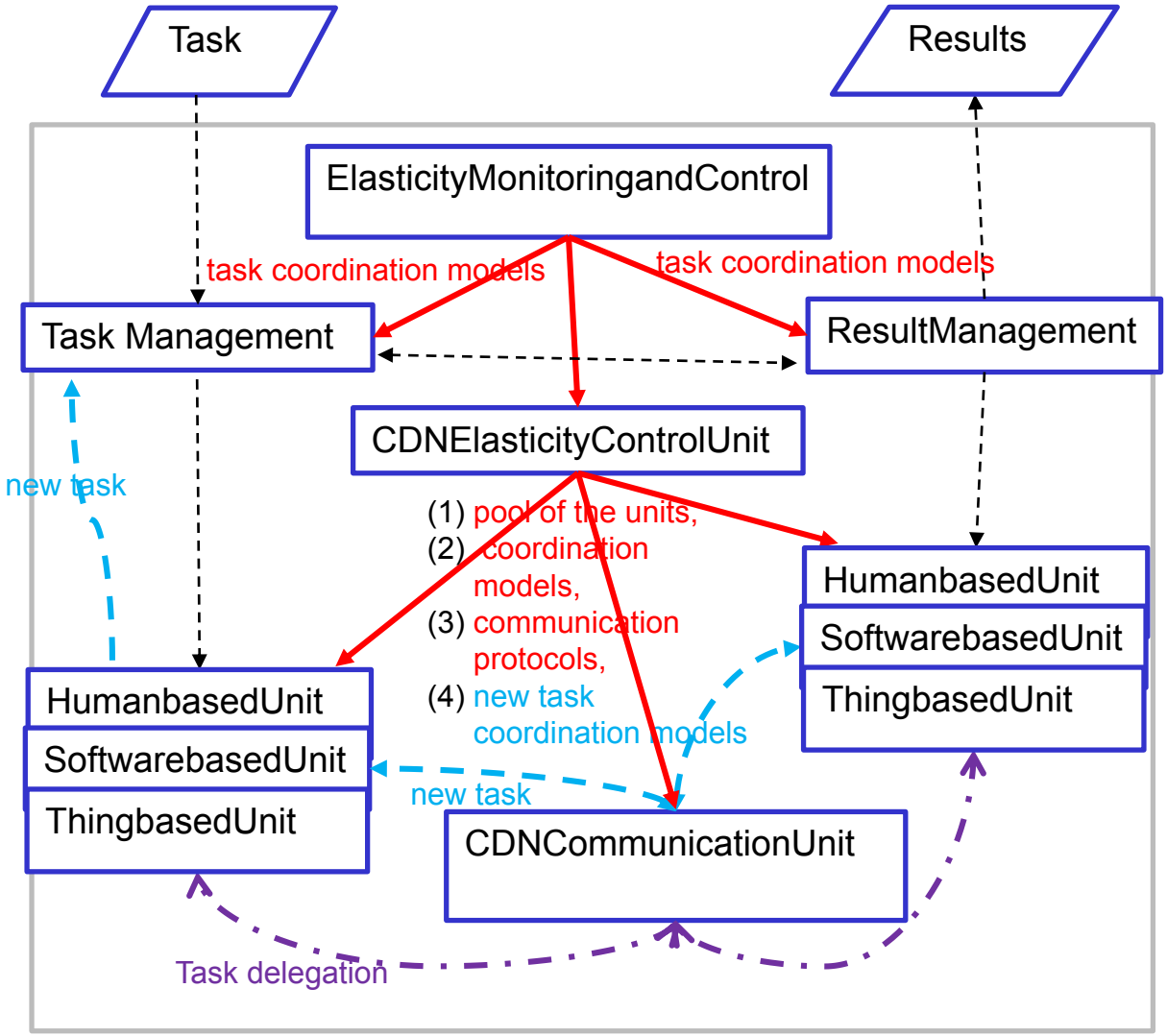
Ctx3: evolving independent task context

- From a task, several independent tasks are created by the collective
- Within or switched from Ctx1/Ctx2
- Multiple types of (automatic/manual) elasticity controls at different places
- Different coordination and unit formation models for evolving independent tasks



Ctx4: evolving dependent task

- Tasks are dependent and the task graph changes (expands/reduces)
- Within Ctx1/Ctx2 or switched from Ctx1/Ctx2/Ctx3
- Multiple types of (automatic/manual) elasticity controls at different places
- Different coordination and unit formation models for evolving dependent tasks



Towards supporting context-aware programming with h²CAS

- SALAM – mainly for quality-aware human-based service formations and task management
  - https://github.com/tuwiendsg/SALAM
- SmartCOM – middleware for HDA-CAS Communication
  - https://github.com/tuwiendsg/SmartCom
- Elasticity Control
  - For software-based services and thing-based services
    - https://github.com/tuwiendsg/rSYBL/
  - For human-based services: currently being developed (based on our work in ICSOC 2013/CAISE 2014)
- Multi-level coordination models
  - To appear in CollaborateCom 2014

Conclusions and future work

- HDA-CAS can be established and changed due to specific context
  - Identify possible contexts for the development of HDA-CAS using hybrid compute units
  - Propose h²CAS and conceptualizing its main blocks
- Future work
  - A h²CAS specification for programming constructs and models
  - Tools and middleware for compiling and deploying and control and provisioning h²CAS

Thanks for your attention!

Hong-Linh Truong

Distributed Systems Group
TU Wien

truong@dsg.tuwien.ac.at
dsg.tuwien.ac.at/research/viecom