A Component-Minded Route from CIM to Real Implementations

And What Tools Can Do for It

88solutions

Manfred R. Koethe
Chief Technology Officer

©2004 88solutions Corp.

First Steps (Buyer – Seller Example)

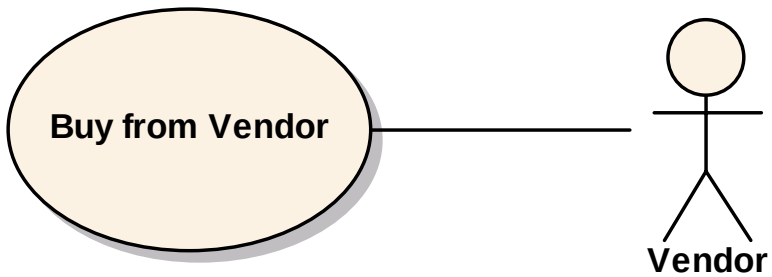
First, develop / discover the Requirements

- Buyer wants to acquire something from Seller
- Seller agrees to provide this something to Buyer
- ➔ Buyer needs an Interface to interact with the Seller
- ➔ Seller needs an Interface to interact with the Buyer
- ➔ An orderly interaction (Protocol) between Buyer and Seller needs to be defined

Use Case Diagrams help to clarify this

©2004 88solutions Corp.

Buyer – Seller Example



Buyer – Seller Relationship from the Customer's viewpoint

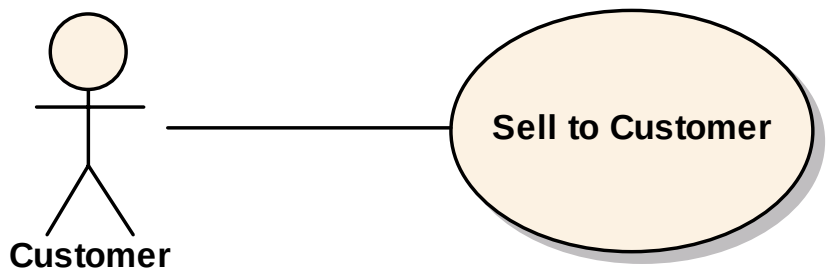
➔The Use Case defines the interface to the Vendor
➔The Vendor is an external entity

©2004 88solutions Corp.

Buyer – Seller Example

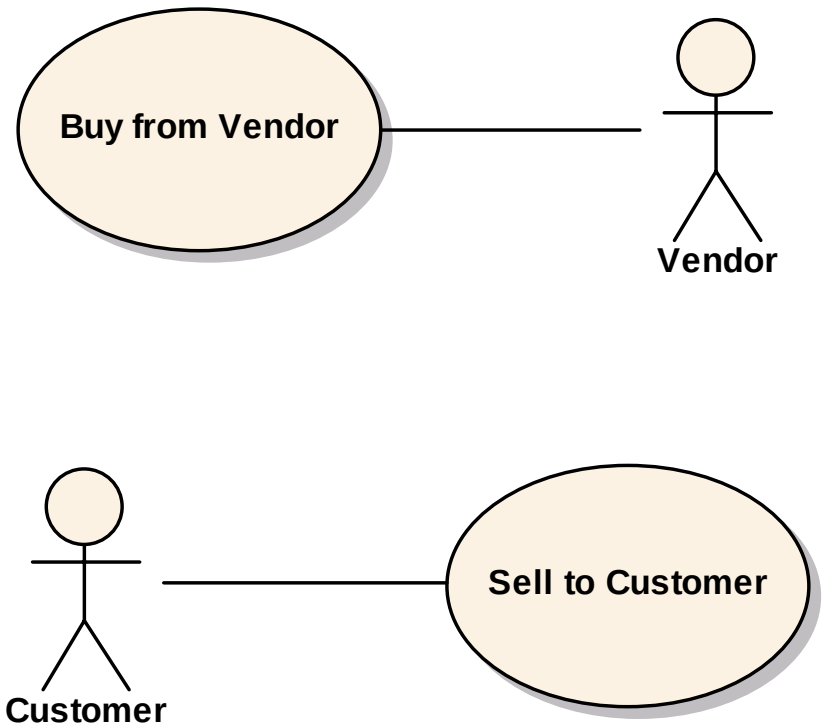
Buyer – Seller Relationship from the Vendor's viewpoint

- ➔The Use Case defines the interface to the Customer
- ➔The Customer is an external entity



©2004 88solutions Corp.

Combined View



©2004 88solutions Corp.

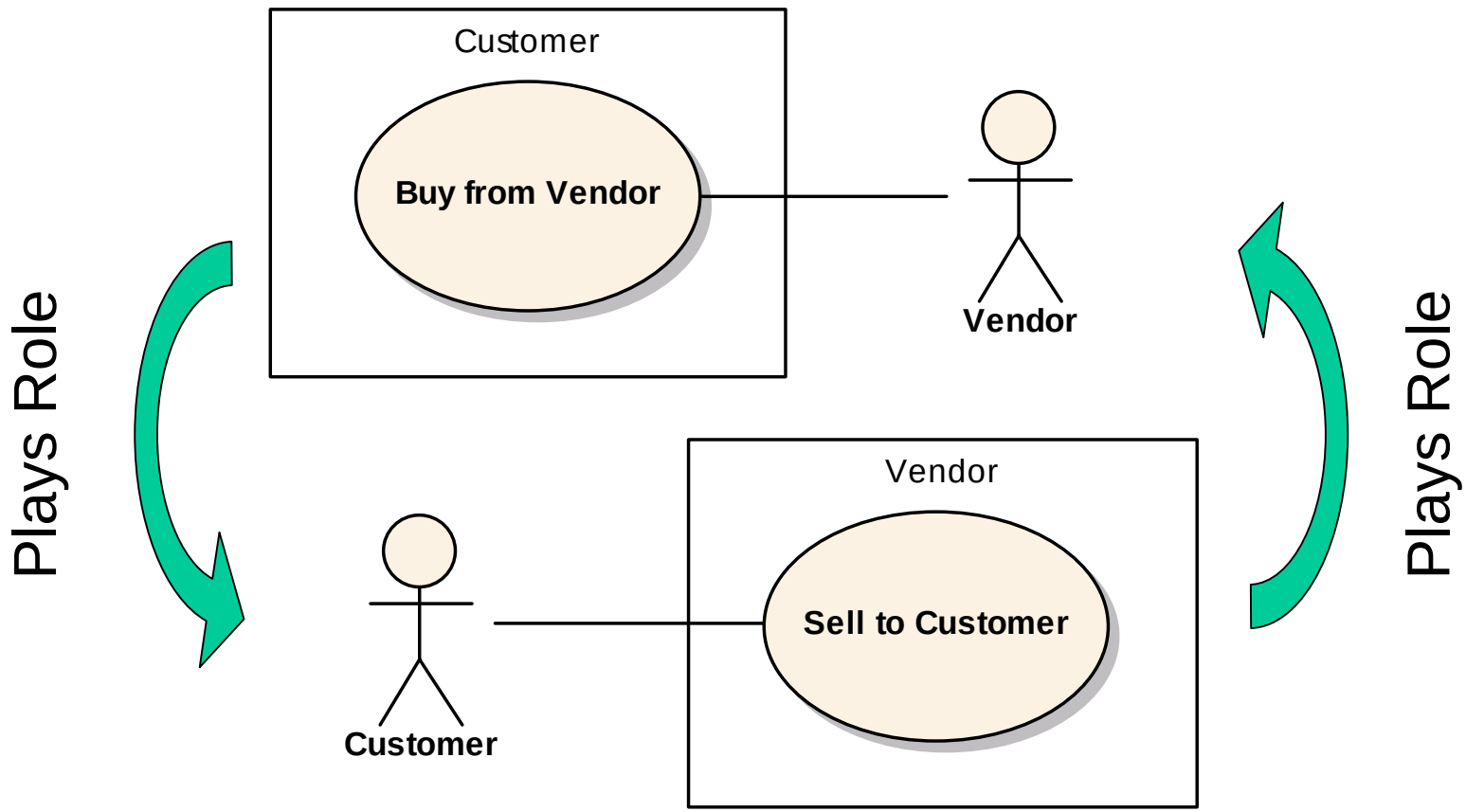
Computational Independent Model

- Nothing so far implied any use of computers
- This is independent from the fact we use UML diagrams to express the model

- It is very important for good modeling results to stay on a high level of abstraction as long as possible
- Therefore, the next step is Discovery of Roles

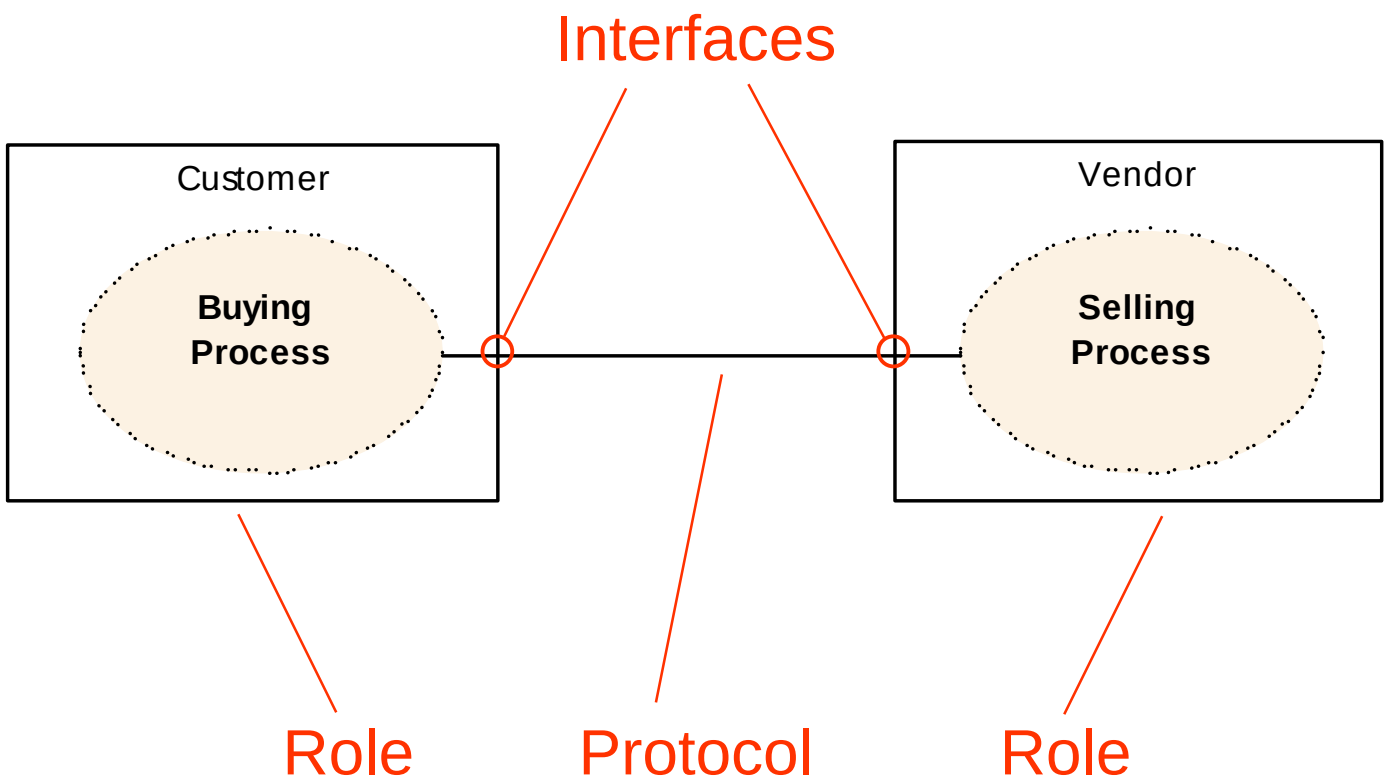
©2004 88solutions Corp.

Discovery of Roles



©2004 88solutions Corp.

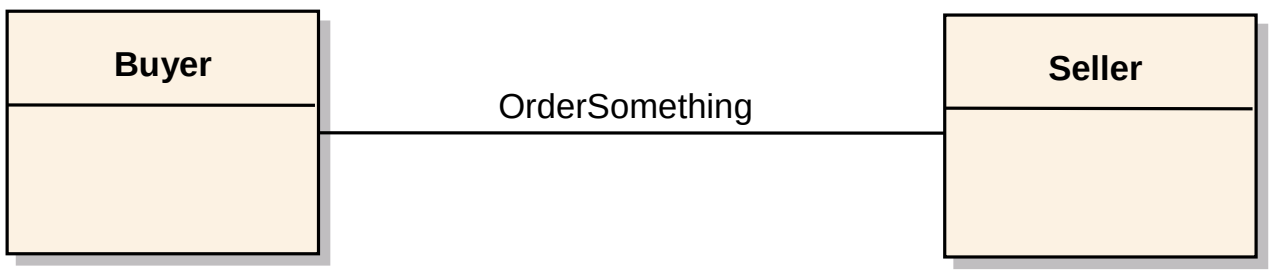
Buyer – Seller Collaboration



©2004 88solutions Corp.

How to model this efficiently?

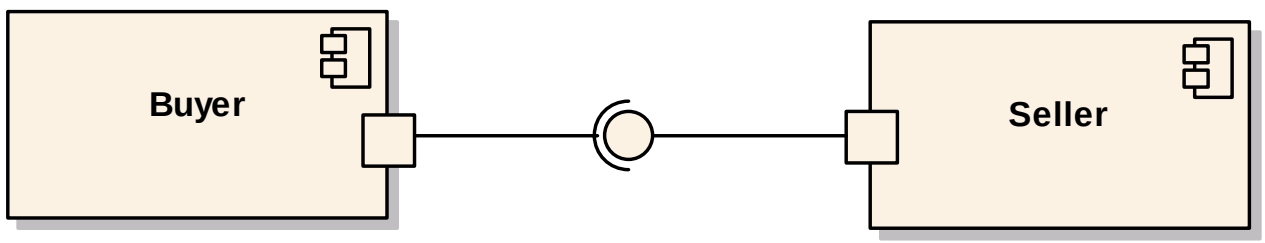
- Plain UML (even UML-2) is not expressive enough to represent the full Buyer – Seller Relationship without excessive extra modeling
- No direct support to refine each role
- No support to express the full semantic of the Buyer – Seller interaction



©2004 88solutions Corp.

How to model this efficiently?

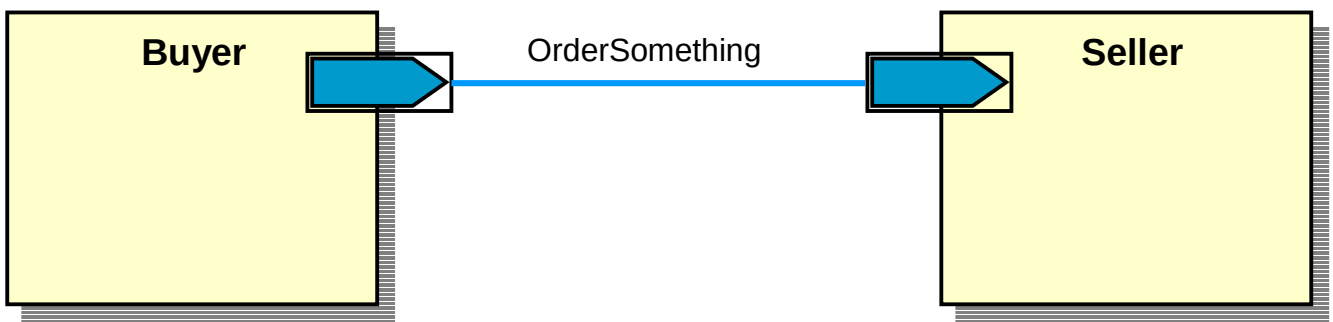
- UML-2 Components add the ability of refinement
- Top level components can be considered as roles
- However, the available mechanisms to specify the Buyer – Seller interaction are still not sufficient



©2004 88solutions Corp.

More Component Power: CCA

- The component Collaboration Architecture (CCA) as defined in the EDOC specification provides sufficient semantic
- The CCA model applies to any component interaction, not only to business components



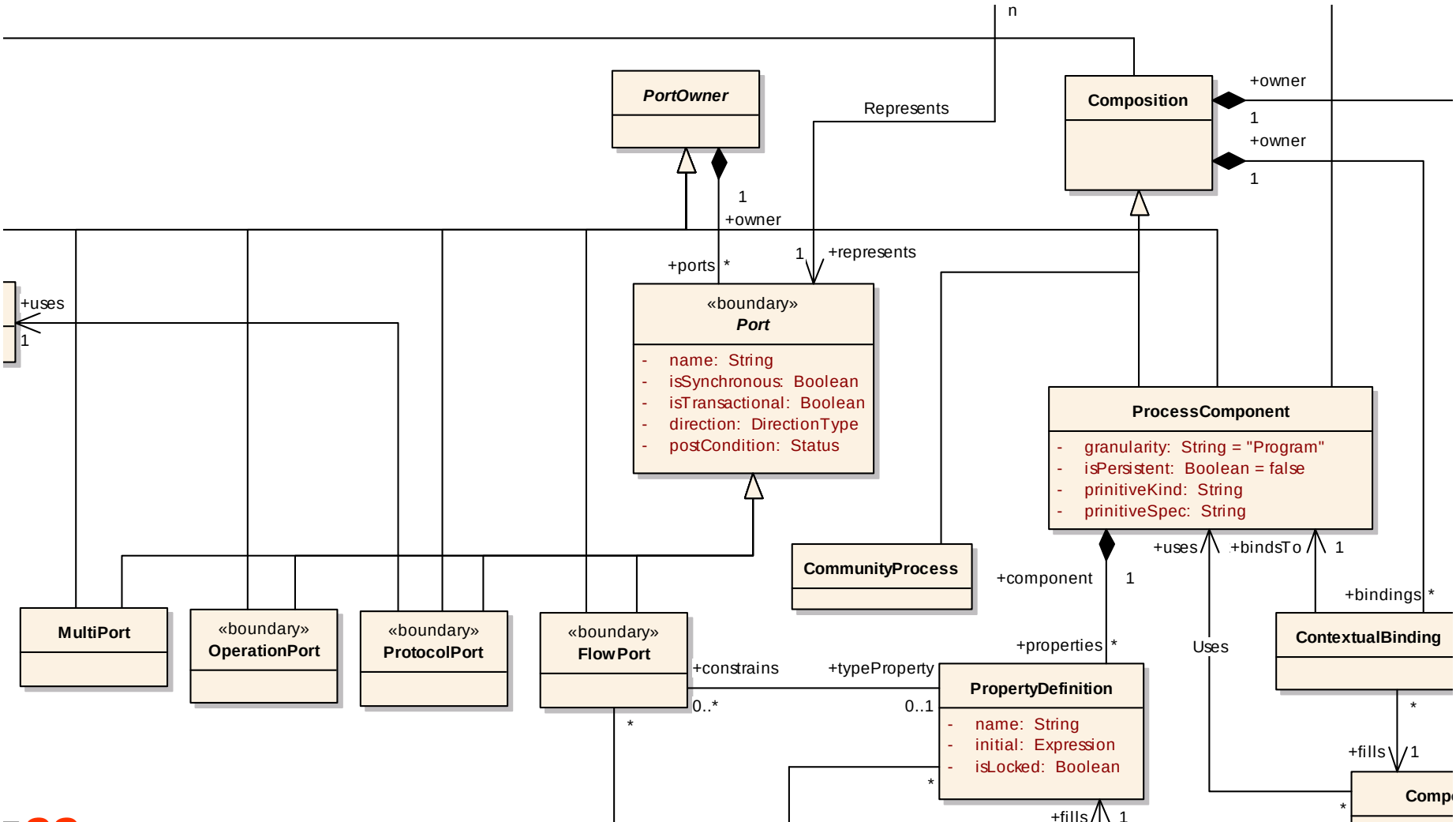
©2004 88solutions Corp.

CCA Port Architecture

- **OperationPort** Simple control flow-like invocations, information transfer via signatures (synchronous and asynchronous)
- **FlowPort** Pure Dataflow interactions, no invocation semantic
- **ProtocolPort** Container for a more complex port architecture, defining a protocol which is realized through embedded ports of any kind
- **MultiPort** Container for the construction of arbitrarily complex ports

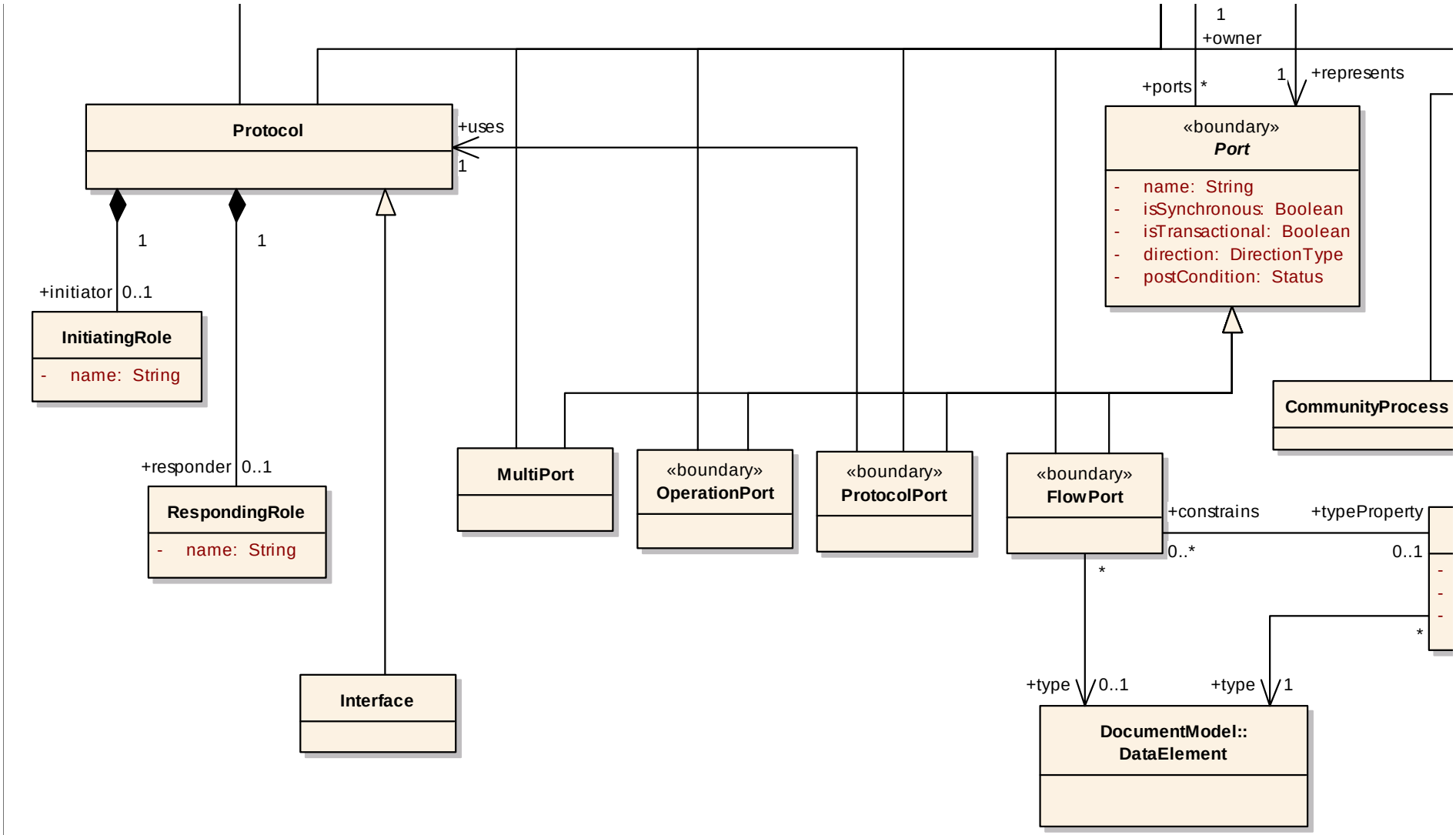
©2004 88solutions Corp.

EDOC / CCA Component Model



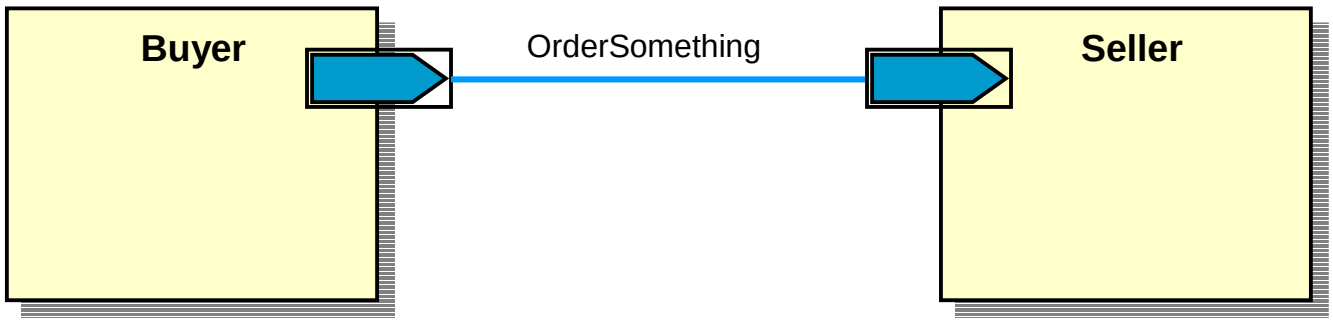
©2004 88solutions Corp.

CCA Component Port Structure



©2004 88solutions Corp.

CCA Component Modeling



Protocol Ports opened:

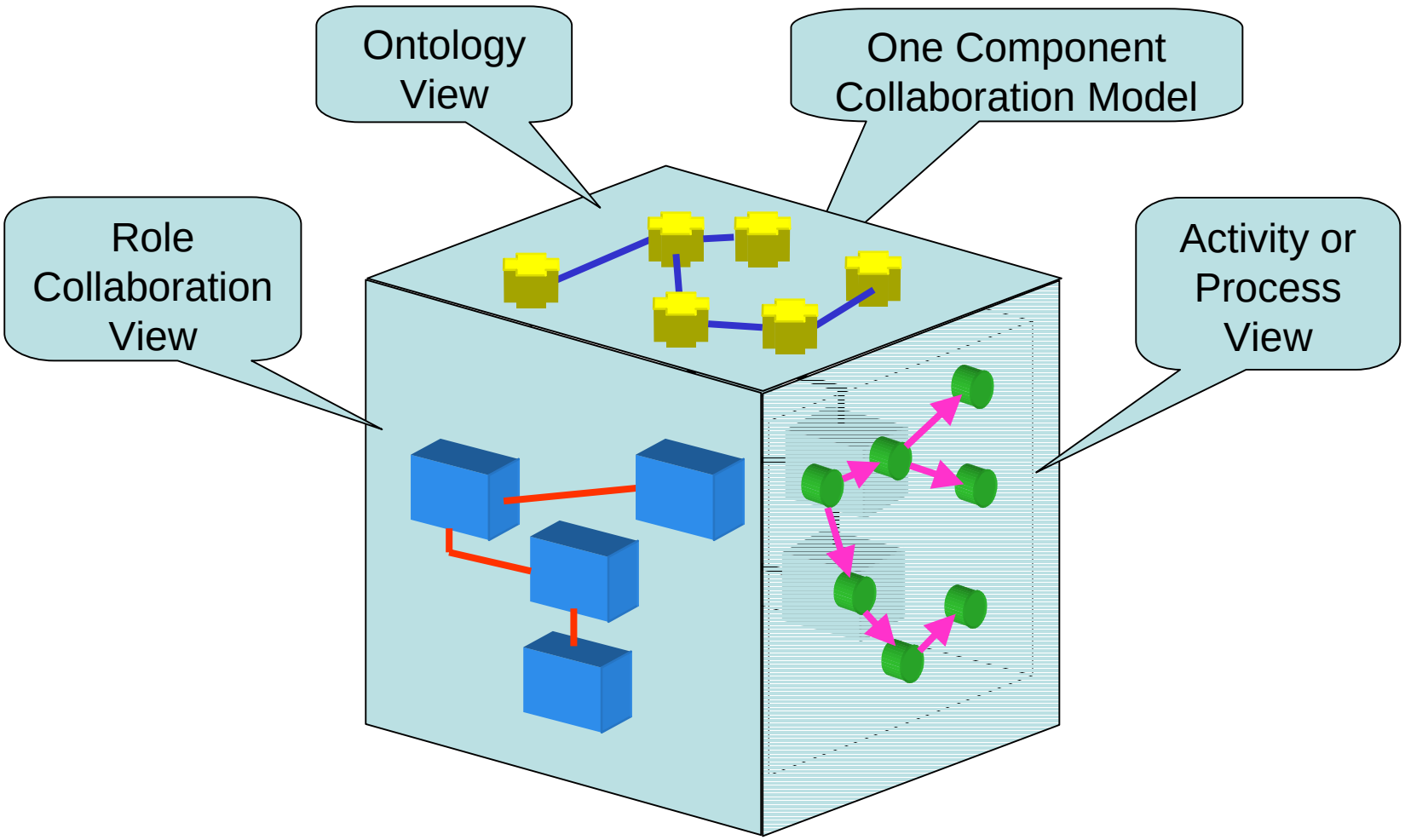
©2004 88solutions Corp.

One Model – Multiple Views

- EDOC specifies already multiple views to the underlying component collaboration model
  - Role Collaboration View
  - Process and Activity View
  - Entity Relationship View
- A combination with the Ontology Definition Metamodel adds one or more Ontology Views
- As a result, very powerful modeling tools can be built, allowing the user to work in multiple views concurrently

©2004 88solutions Corp.

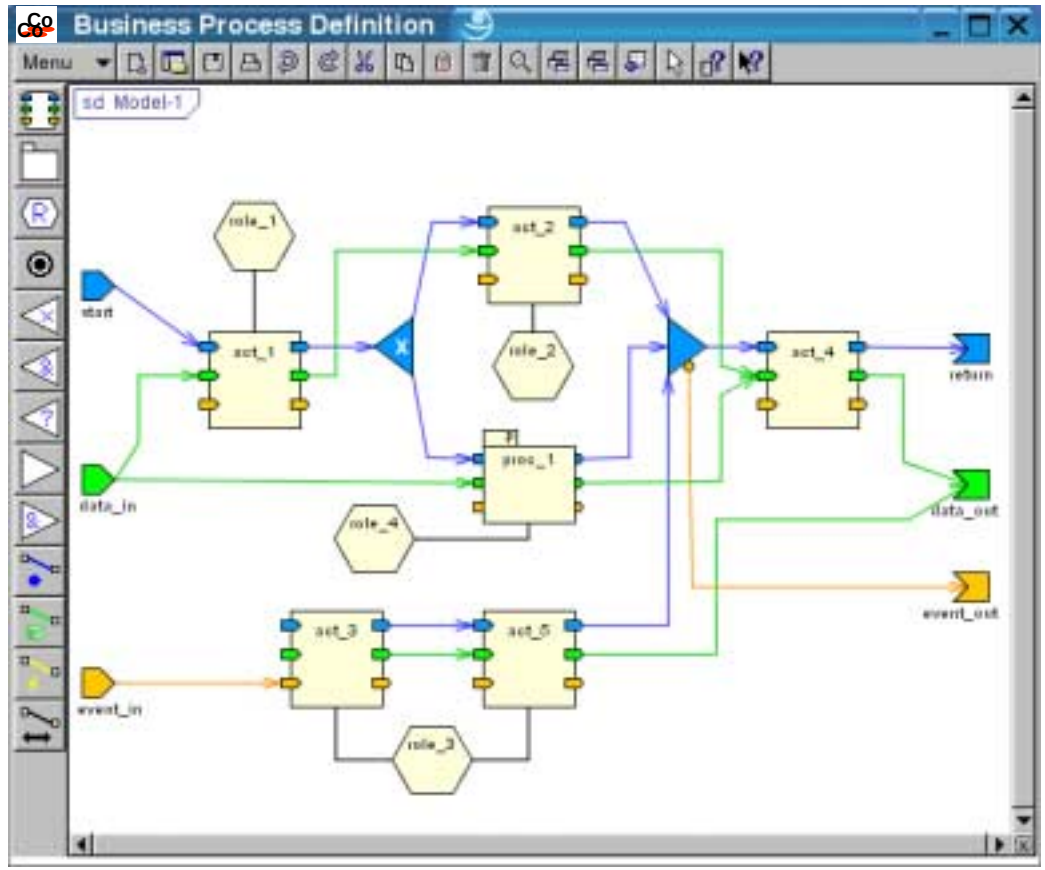
Role and Business Process View



©2004 88solutions Corp.

Tool Example: Process View

- Processes are composed from Activities
- Activities are provided by Components
- Roles provide abstract representations for Components
- Color helps to distinguish Choreography, Data and Events



©2004 88solutions Corp.

To become real: Deployment

- Components are implemented by Artifacts
- Artifacts are performed by Performers
- Performers finally map to an execution Domain, composed from a network of execution Nodes
- Required deployment information can be derived from the component descriptions
- The “Deployment and Configuration” specification defines the full mechanism

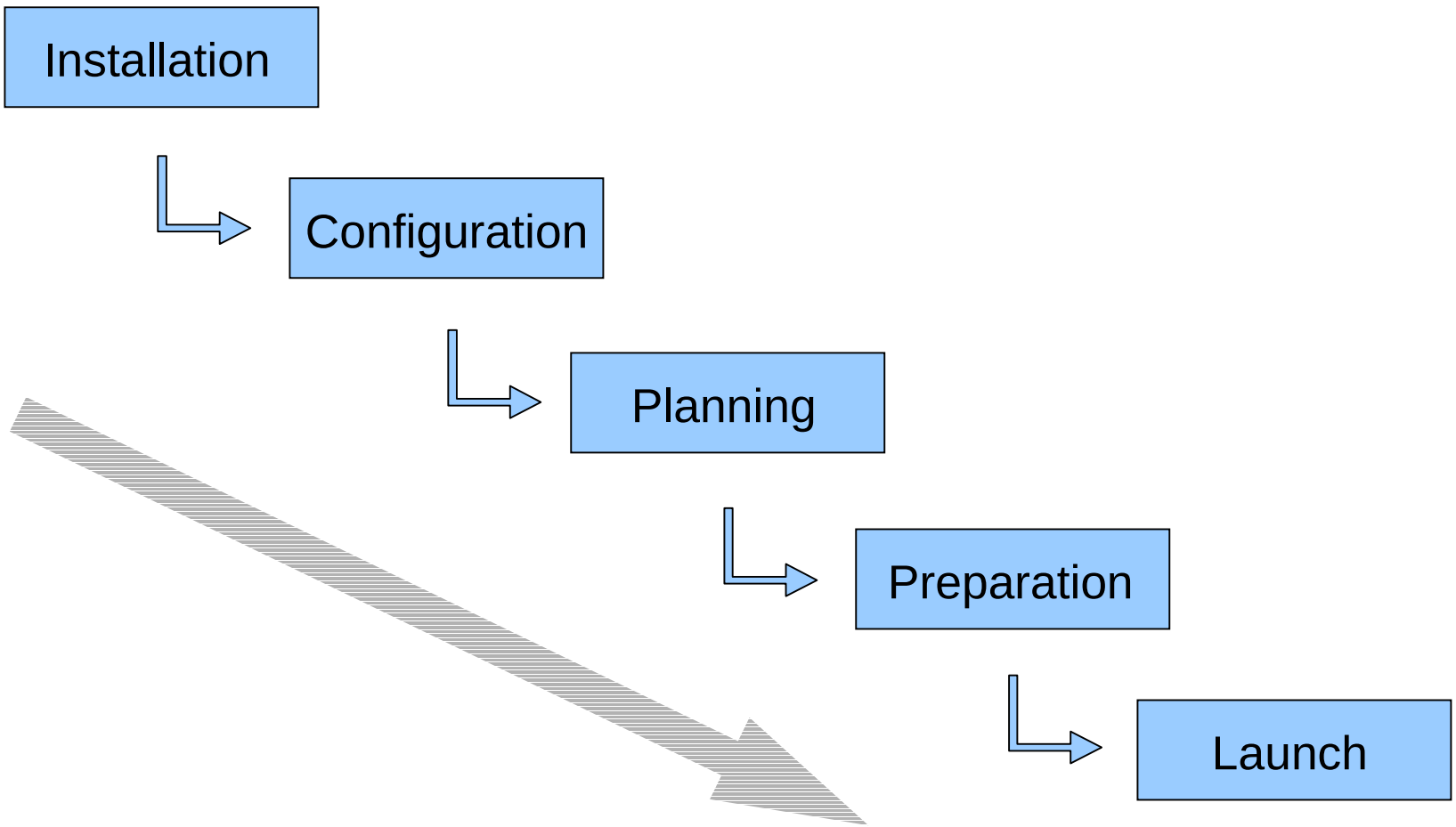
©2004 88solutions Corp.

To become real: Map to the PSM

- With increasing component refinement, the CIM becomes a PIM
- After selection of the target platform and technology, we are ready to map to the PSM
  - J2EE
  - CORBA Components
  - ...
- The mapping is typically straight forward and can be handled during the Deployment and Configuration process

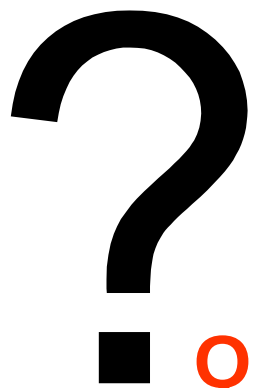
©2004 88solutions Corp.

Deployment & Configuration



©2004 88solutions Corp.

Thank You ...



Questions?

©2004 88solutions Corp.