A Framework to Dynamically Manage Distributed Virtual Environments
Objectives

- Full distribution
- Scalability
- Robustness
- Simplicity
- Spatial continuity
- Dynamic management
Our proposal

• Rely on group communication
• Collaborative kernels approach:
  – Virtual machine
  – Renderer
  – Group Communication module
• Language-based architecture
Overview of the architecture

- **NET**
  - Ensemble

- **KERNEL**
  - OCaml

- **DISPLAY**
  - OpenGL

**GOAL**
The GOAL language...

...should make it easy to

- represent 3D data and behavior
- process network and user events
- manage distribution
GOAL characteristics

- Object-oriented
- Message-oriented
- Event-driven

Group & Object
Asynchronous Language
@Class Shape =
{
    Inherit Object;
    Implements IShape with shape;
    Initialize with
    {...};
    static string file3DS with
    {Self <- create3DS Self.file3DS;
     @Self <- Set isReady = true;
     @Self <- Set fileName = Self.file3DS;}
    ...
    bool isReady;
    string fileName;
    extern ExtShape shape;
};
Dynamic world modification

@Avatar_Of_demo <- Set viewRadius = 500.0;

@shape_castle = New Shape;
@shape_castle <- Set file3DS = "castle.3ds";
@castle = New GridMemberObject3d;
@castle <- Set shape = shape_castle;
@castle <- Set position = < -100. 102. 569. >;
@castle <- Set inGrid = grid_dune;
@castle <- Set rotation = < 0. -70. 0. >;
@castle <- Set scale = < 30. 30. 30. >;

before

after
Introspection

Instance dumping

@castle <- dump;
>castle = { shape = shape_castle,
  isShapeReady = true,
  inGrid = grid_dune,
  position = <-100.2 102.6 569.0>,
  rotation = <0.0 -70.0 0.0>,
  scale = <30.0 30.0 30.0>,...}
Conclusion & future work

- Pros
  - good experimentation framework
  - language-based
  - group communication

- Cons
  - GOAL expressivity
  - security and authentication features
  - synchronization process
Introspection

Class dumping

@Goal <- dump_class "Kernel";
>Kernel (inherit=Object) =
{
    bool @initialized with {<daemon code>};
    ...
    setPasswd = set_passwd;
    setUser = set_user;
    ...
    private echo_callback = echo_callback;...};