

B IDL and VPL Definitions for the conference example

This appendix contains the complete IDL and VPL definitions for the conference example in chapter 6.

B.1 IDL

```
module demo {  
  
    module conference {  
  
        /* simple Documents... */  
  
        interface Document  
        {  
            /* the title is set on creation */  
            readonly attribute string title;  
            readonly attribute string author;  
  
            string read();  
            void write( in string text );  
            void append( in string text );  
            void update( in string text );  
        };  
  
        interface Review; // forward decl.  
  
        /* Papers are created using the Submission interface */  
  
        typedef long ReviewerId;  
        typedef sequence< ReviewerId > ReviewerIdSeq;  
  
        interface Paper : Document  
        {  
            /* id, assigned when created by Submission */  
            readonly attribute long number;  
        };  
    };  
};
```

```
void submit();

Review submitReview( in string review,
                    in ReviewerId reviewer);
ReviewerIdSeq listReviews();

Review getReview( in ReviewerId reviewer);
};

interface Review : Document
{
    readonly attribute ReviewerId reviewer;
};

struct PaperId
{
    long number;
    string title;
};

typedef sequence <PaperId> PaperIdSeq;

exception AlreadyRegistered {};
exception IncompleteInformation {};

interface SubmissionManagement
{
    /* for authors */
    Paper registerPaper( in string author_name,
                       in string title )
        raises( AlreadyRegistered, IncompleteInformation );

    /* for reviewers */
    PaperIdSeq listPapers();
    Paper getPaper( in long paperNumber );
};

interface ConferenceManagement
{
    /* retrieve submission management interface */

    SubmissionManagement getSubmissionManagement();
    readonly attribute Document callForPapers;

    /* conference management operations */
};
```

```

        void issueCallForPapers( in string cfp );
        void beginSubmission();
        void deadlineReached();
        void makeDecision();
    };
};
};

```

B.2 VPL

```

policy Conference
{
    roles

        Reviewer
            holds Member
                on "IDL:demo/conference/SubmissionManagement:1.0"
            holds SubmissionAccessing
                on "IDL:demo/conference/ConferenceManagement:1.0"
            holds PaperReviewing
                on "IDL:demo/conference/Paper:1.0"

        Chair: Reviewer
            holds Steering
                on "IDL:demo/conference/ConferenceManagement:1.0"
            holds PaperReading
                on "IDL:demo/conference/Paper:1.0"
            maxcard 1
            excludes Author

        Author
            holds SubmissionAccessing
                on "IDL:demo/conference/ConferenceManagement:1.0"
            holds Registering
                on "IDL:demo/conference/SubmissionManagement:1.0"

    virtual view SubmissionPhase
    virtual view ReviewingPhase

    // initial views

    view Member
        controls "IDL:demo/conference/SubmissionManagement:1.0"
    {
        allow
    }
}

```

```
        listPapers
        getPaper
    }

view SubmissionAccessing
    controls "IDL:demo/conference/ConferenceManagement:1.0"
    {
        allow
            getSubmissionManagement
            callForPapers
    }

view Steering : SubmissionAccessing
    controls "IDL:demo/conference/ConferenceManagement:1.0"
    restricted_to Chair
    {
        allow
            issueCallForPapers
            beginSubmission
            deadlineReached
            makeDecision
    }

view Registering
    controls "IDL:demo/conference/SubmissionManagement:1.0"
    requires SubmissionPhase
    {
        allow
            registerPaper
    }

view Reading controls "IDL:demo/conference/Document:1.0" {
    allow
        read
        title
    }

view Modifying : Reading
    controls "IDL:demo/conference/Document:1.0"
    {
        allow
            update
            write
            append
    }
}
```

```
view PaperReading : Reading
  controls "IDL:demo/conference/Paper:1.0"
  {
    allow
      number
      author
      listReviews
  }

view PaperSubmitting : PaperReading
  controls "IDL:demo/conference/Paper:1.0"
  requires SubmissionPhase
  {
    allow
      submit
  }

view PaperReviewing : PaperReading
  controls "IDL:demo/conference/Paper:1.0"
  restricted_to Reviewer
  requires ReviewingPhase
  {
    allow
      listReviews
      submitReview
  }

view NoMorePaperReviewing
  controls "IDL:demo/conference/Paper:1.0"
  {
    deny
      submitReview
  }

view ReviewAccess
  controls "IDL:demo/conference/Paper:1.0"
  restricted_to Reviewer
  {
    allow
      getReview
  }

view ReviewReading : Reading
  controls "IDL:demo/conference/Review:1.0"
  {
    allow
      reviewer
  }
```

```
schema SubmissionSchema
  observes "IDL:demo/conference/SubmissionManagement:1.0"
  {
    registerPaper
      assigns PaperSubmitting, Modifying on result to caller
  }

schema PaperSchema
  observes "IDL:demo/conference/Paper:1.0"
  {
    submitReview
      assigns
        Modifying on result to caller
      assigns
        ReviewReading on result to Reviewer
      assigns
        ReviewAccess on this to caller
      assigns
        NoMorePaperReviewing on this to caller
    submit
      assigns
        PaperReading on this to Reviewer
      removes
        Modifying on this from caller
  }

schema Steering
  observes "IDL:demo/conference/ConferenceManagement:1.0"
  {
    issueCallForPapers
      assigns
        Reading
          on this.callForPapers
          to Author, Chair, Reviewer
    beginSubmission
      assigns
        SubmissionPhase
          on "IDL:omg.org/CORBA/Object:1.0"
          to Author
    deadlineReached
      assigns
        ReviewingPhase
          on "IDL:omg.org/CORBA/Object:1.0"
          to Reviewer
      removes
        SubmissionPhase
  }
```

```
        on "IDL:omg.org/CORBA/Object:1.0"  
        from Author  
    makeDecision  
        removes  
            ReviewingPhase  
            on "IDL:omg.org/CORBA/Object:1.0"  
            from Reviewer  
    }  
}
```

