Summary of CLIPCM Concerns
October 7, 1991

- BINDING MODEL
  - Identification of entry points
    - Not yet invocation of the called procedure
  - Association between objects
  - **Status in CLIPCM** : Currently does not exist in the CLIPCM.
  - **Relation to other concerns** : State
- CONTROL FLOW
  - Synchronous
    - Single thread of execution
  - Asynchronous
    - Multiple threads of execution
  - Recursion
  - Call Backs
  - Exception handling
  - **Status in CLIPCM** : The issue of asynchronous control flow is not prohibited in the CLIPCM, but it is not explicitly defined. An initial cut at a model for exception handling exists. Call backs are not addressed except for in the IDN text.
  - **Relation to other concerns** : Call management, completion status
- PARAMETER PASSING MECHANISMS
  - Call by reference
    - Dereference on access of a procedure parameter
    - System automatically does dereference
    - Updated value immediately reflected to caller
    - Special case of passing a pointer
  - Call by value
    - IN only (immutable?), copy of argument value supplied to called routine
  - Call by name
    - THUNK variety where the IDN expression is evaluated
      - THUNK is a pseudo procedure as in Algol-68
    - Crude call by value
  - Call by need
    - Another THUNK variety where the IDN expression is not evaluated
    - Just pass the THUNK

        - Additional issues relating to first need and every need should be addressed
    - Call by value return
        - Copy IN, copy OUT
        - Possibly an In only parameter associated with an OUT only parameter
    - Call by value passing pointer
        - Access of procedure parameter gets pointer value
        - User must do the dereference
    - Call by result
        - OUT only
    - Call by value optional
        - Basically by value, although certain parameters may be marked as being optional (i.e., x(a,,c))
    - Call by value default
        - Similar to by value optional, but the optional value defaults to something
    - Call by value optional return
        - Basically by value return with certain parameters marked as optional
    - Interface storage management
        - ECMA-PCTE discussions
    - **Status in CLIPCM** : The mechanisms Value, Value-Return, and Reference are currently in version 3.0. In addition to the missing mechanisms, another section dealing with "aliasing" needs to be included in the CLIPCM. Also, the issue of mutable/immutable is closely related to parameter passing.
    - **Relation to other concerns** : State, environment sharing
- COMPLETION STATUS
    - Normal return from called program
    - Exceptional return from called program
        - Exceptional completion status from called program
        - Exception raising
            - Caller decided to signal for called program termination
    - **Status in CLIPCM** : The CLIPCM recognizes different types of completion forms from the called procedure. Exception raising is not addressed.
    - **Relation to other concerns** : Call management and control flow.
- CALL MANAGEMENT
    - Cancellation of call
        - Not by server or client procedure
        - Pending Call
            - Exception

  - **Status in CLIPCM** : Not currently included in the CLIPCM and probably does not belong there. This is an operating system level concern and is outside the scope of the CLIPCM.
  - **Relation to other concerns** : Completion Status, Control Flow, also related to POSIX runtime extensions.

- STATE
  - Execution environment
    - When created
    - How long in existence
    - Environment initialization
      - Body, Parameter, State (Context handle issue)
    - Environment identified on subsequent calls
      - Instance identification
      - Instantiation of procedure body
      - Instance of parameters
  - **Status in CLIPCM** : Environment initialization is currently implementation defined.
  - **Relation to other concerns** : Environment sharing, parameter passing mechanisms
- ENVIRONMENT SHARING
  - Non-Local References
    - Global data, common blocks
    - Passing pointer & procedure parameters
      - Dynamic scoping
      - THUNK (pass procedure and environment)
      - dynamic name binding
        - table model, table shared with called procedure
  - **Status in CLIPCM** : Global data is addressed in the CLIPCM as an implementation defined feature. Appendix A provides a description of procedure parameters.
  - **Relation to other concerns** : State
- PARAMETER TYPES
  - Types are defined in the CLIDT
  - **Status in CLIPCM** : All data types defined in CLIDT
  - **Relation to other concerns** : State, Parameter passing mechanisms