

# ISO/IEC 13211-2:2000-06 (E)

## Information technology - Programming languages; Prolog - Part 2: Modules

---

<b>Contents</b>		<b>Page</b>
Foreword .....		v
Introduction .....		vi
<b>1</b>	<b>Scope 1 1.1 Notes .....</b>	<b>1</b>
<b>2</b>	<b>Normative reference 1 3 Terms and definitions 1 4 Compliance 3 4.1 Prolog processor .....</b>	<b>3</b>
<b>4.2</b>	<b>Module text .....</b>	<b>3</b>
<b>4.3</b>	<b>Prolog goal .....</b>	<b>3</b>
<b>4.4</b>	<b>Prolog modules .....</b>	<b>3</b>
<b>4.4.1</b>	<b>Prolog text without modules .....</b>	<b>3</b>
<b>4.4.2</b>	<b>The module user .....</b>	<b>4</b>
<b>4.5</b>	<b>Documentation .....</b>	<b>4</b>
<b>4.5.1</b>	<b>Dynamic Modules .....</b>	<b>4</b>
<b>4.5.2</b>	<b>Inaccessible Procedures .....</b>	<b>4</b>
<b>5</b>	<b>Syntax 4 5.1 Module text .....</b>	<b>4</b>
<b>5.2</b>	<b>Terms .....</b>	<b>4</b>
<b>5.2.1</b>	<b>Operators .....</b>	<b>4</b>
<b>6</b>	<b>Language concepts and semantics 4 6.1 Related terms .....</b>	<b>5</b>
<b>6.1.1</b>	<b>Qualified and unqualified terms .....</b>	<b>5</b>
<b>6.2</b>	<b>Module text .....</b>	<b>5</b>
<b>6.2.1</b>	<b>Module user .....</b>	<b>5</b>
<b>6.2.2</b>	<b>Procedure Visibility .....</b>	<b>5</b>
<b>6.2.3</b>	<b>Module interface .....</b>	<b>5</b>
<b>6.2.4</b>	<b>Module directives .....</b>	<b>6</b>
<b>6.2.5</b>	<b>Module body .....</b>	<b>7</b>
<b>6.2.6</b>	<b>Clauses .....</b>	<b>7</b>
<b>6.3</b>	<b>Complete database .....</b>	<b>8</b>
<b>6.3.1</b>	<b>Visible database .....</b>	<b>8</b>
<b>6.3.2</b>	<b>Examples .....</b>	<b>8</b>
<b>6.4</b>	<b>Context sensitive predicates .....</b>	<b>8</b>
<b>6.4.1</b>	<b>Metapredicate built-ins .....</b>	<b>8</b>
<b>6.4.2</b>	<b>Context sensitive built-ins .....</b>	<b>9</b>
<b>6.4.3</b>	<b>Module name expansion .....</b>	<b>9</b>
<b>6.4.4</b>	<b>Examples: Metapredicates .....</b>	<b>9</b>
<b>6.5</b>	<b>Converting a term to a clause, and a clause to a term .....</b>	<b>10</b>
<b>6.5.1</b>	<b>Converting a term to the head of a clause .....</b>	<b>10</b>
<b>6.5.2</b>	<b>Converting a module qualified term to a body .....</b>	<b>10</b>
<b>6.5.3</b>	<b>Converting the body of a clause to a term .....</b>	<b>11</b>
<b>6.6</b>	<b>Executing a Prolog goal .....</b>	<b>12</b>
<b>6.6.1</b>	<b>Data types for the execution model .....</b>	<b>12</b>
<b>6.6.2</b>	<b>Initialization .....</b>	<b>12</b>
<b>6.6.3</b>	<b>Searching the complete database .....</b>	<b>13</b>
<b>6.6.4</b>	<b>Selecting a clause for execution .....</b>	<b>13</b>
<b>6.6.5</b>	<b>Backtracking .....</b>	<b>14</b>
<b>6.6.6</b>	<b>Executing a user-defined procedure: .....</b>	<b>14</b>
<b>6.6.7</b>	<b>Executing a built-in predicate .....</b>	<b>14</b>
<b>6.7</b>	<b>Executing a control construct .....</b>	<b>14</b>
<b>6.7.1</b>	<b>call/1 .....</b>	<b>14</b>

6.7.2	catch/3 .....	15
6.7.3	throw/1 .....	15
6.8	Predicate properties .....	16
6.9	Flags .....	16
6.9.1	Flag: colon sets calling context .....	16
6.10	Errors .....	16
6.10.1	Error classification .....	16
7	<b>Built-in predicates 16 7.1 The format of built-in predicate definitions .....</b>	<b>16</b>
7.1.1	Type of an argument .....	16
7.2	Module predicates .....	16
7.2.1	current module/1 .....	17
7.2.2	predicate property/2 .....	17
7.3	Clause retrieval and information .....	18
7.3.1	clause/2 .....	18
7.3.2	current predicate/1 .....	19
7.4	Database access and modification .....	20
7.4.1	asserta/1 .....	20
7.4.2	assertz/1 .....	21
7.4.3	retract/1 .....	21
7.4.4	abolish/1 .....	22