proposed move to System F

At the Manticore meeting on 11/7/07, we discussed moving BOM in the direction of System F.

The following summarize the proposed changes to the <code>BOM</code> module:

New syntactic forms:

- let x : tau = y [tauVec]
- apply f [tauVec] (... / ...)
- let x : tau = CONS [tauVec] (... )
- let x : tau = CONST [tauVec]
- throw k [tau] (...)

New types:

- alpha (* type variables *)
- forall alphaVec . tau
- T [tauVec]
- tau + tau

Here are the changes to the datatypes in the <code>BOM</code> module:

datatype term = ...

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>E_Apply</td>
</tr>
<tr>
<td>E_Throw</td>
</tr>
<tr>
<td>E_HLOp</td>
</tr>
</tbody>
</table>

and rhs = ...

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>E_TyApply</td>
</tr>
<tr>
<td>E_DCon</td>
</tr>
</tbody>
</table>

and lambda = FB of

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown macro: { f }</td>
</tr>
</tbody>
</table>

and pat = P_DCon of (data_con * ty list * var list)

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>P_DCon</td>
</tr>
</tbody>
</table>

and the changes to the BOMTy module:

datatype ty = T_Var of ty_var

<table>
<thead>
<tr>
<th>Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>T_Var</td>
</tr>
<tr>
<td>T_Forall</td>
</tr>
<tr>
<td>T_TyCon</td>
</tr>
</tbody>
</table>

We'll also need to add the <code>ty_var</code> type and make changes to the representation of <code>tyc</code>s.