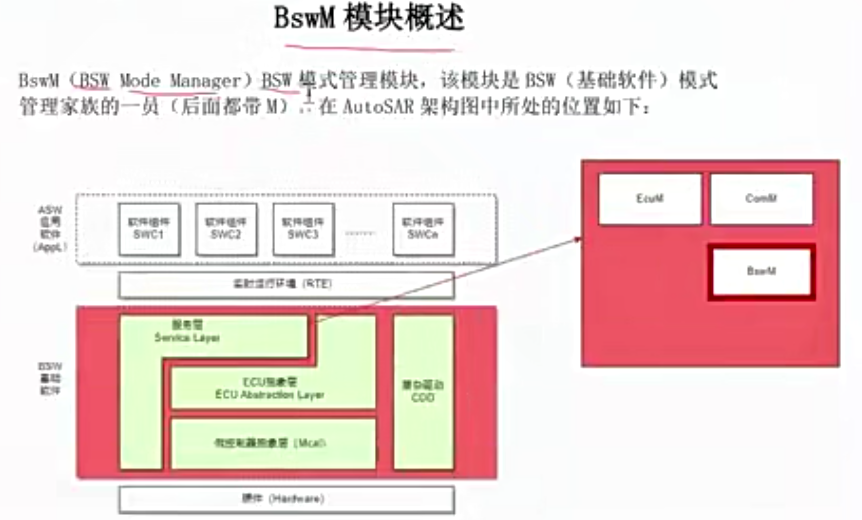
# Autosar BswM模块讲解

**BswM模块概述**

BswM(BSW Mode Manager)BSW模式管理模块，该模块是BSW(基础软件)模式管理家族的一员（后面都带M），在AutoSAR架构图中所处的位置如下：



AutoSAR基础软件有几个“管家”：

BswM：BSW Mode Manager（BSW模式管理），BSW模式的总管，**注意是BSW模式**

ComM:Comunication Manager（通信管理），通信总管

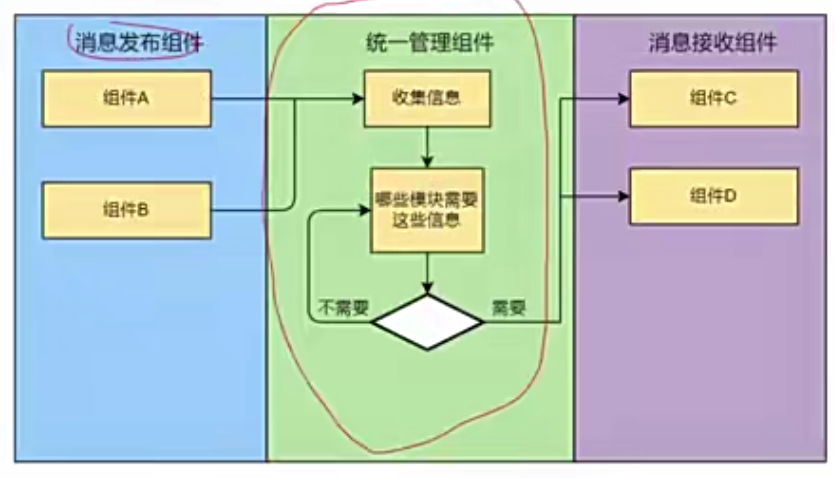
EcuM:Ecu Manager(Ecu管理)，Ecu总管

这期主要讲解BswM。

它的职责是根据简单的规则对来自应用层SW-C或其他BSW模块的模式请求进行仲裁，并根据仲裁结果执行动作。

说的简单点，AutoSAR架构中模块非常多，模块之间可能存在一个模块触发了某个条件，会触发另一个模块做相关的动作，而且这种场景在AutoSAR中非常多(比如：EcuM那边上电流程走完了，就需要ComM开启全通信)。这就需要一个统一模块，来对这些条件和动作进行管理，达到软件解耦的要求。

也可以把BswM理解成整个AutoSAR的信息搜集中心，静态配置好条件和动作，将两个模块关联起来。当BswM监控该条件满足的时候，就会通知到相关模块。（比如:EcuM那边上电流程完成，触发条件，BswM通知到ComM，ComM开启全通信）。



这里统一管理组件就是BswM模块。

1. BswM条件输入来源于SWC或者BSW模块
2. BSWM内部会进行条件判断，是否满足某个规则
3. 满足规则后执行相应的Action

**BswM模块实现细节**

BswM主要分成两大部分，Mode Arbitration(模式仲裁)或者Mode Control（模式控制）。

模式仲裁：

Bsw会接收到SWC或者BSW模块的Mode Request（模式请求）或者Mode Indication（模式通知）作为模式仲裁的两种输入方式。

比如

DCM接收到关闭正常通信的诊断请求

DCM向BswM发送Mode Request(模式请求)

BswM根据模式规则进行模式仲裁，并执行相应的Action，通知ComM进行No Comunication.



模式仲裁过程

模式请求来源（Mode Source）：

模式请求（mode request）

模式指示（mode indications）

模式切换通知（mode switch notification）

时间请求（event request）

时间清楚请求（clearing event request）

**仲裁规则（Arbitration Rules）**

规则是由一组模式请求条件组成的逻辑表达式。当输入模式请求和模式指示改变时，或者在BSWM主函数的执行期间，进行规则判断。判断的结果（真或假）用于决定相应模式控制动作列表的执行。



**模式条件（ModeCoodition）和逻辑表达式（Logical Expressices）**：

**模式条件**指的是根据模式请求来源与静态设定的值相比较的**单一表达式**。比如若模式请求来源为N，N为设定值，那么模式条件就是相似于M = N或者M ！= N这样单一的表达式。

**逻辑表达式**通过使用不同逻辑运算符（如AND、OR、XOR、NOT和NAND），可以实现多个模式条件的逻辑组合。例如：Logic Expression X需要两个Mode Indication1 与Mode Indication2的逻辑组合：

Mode Condition 1： A == 1

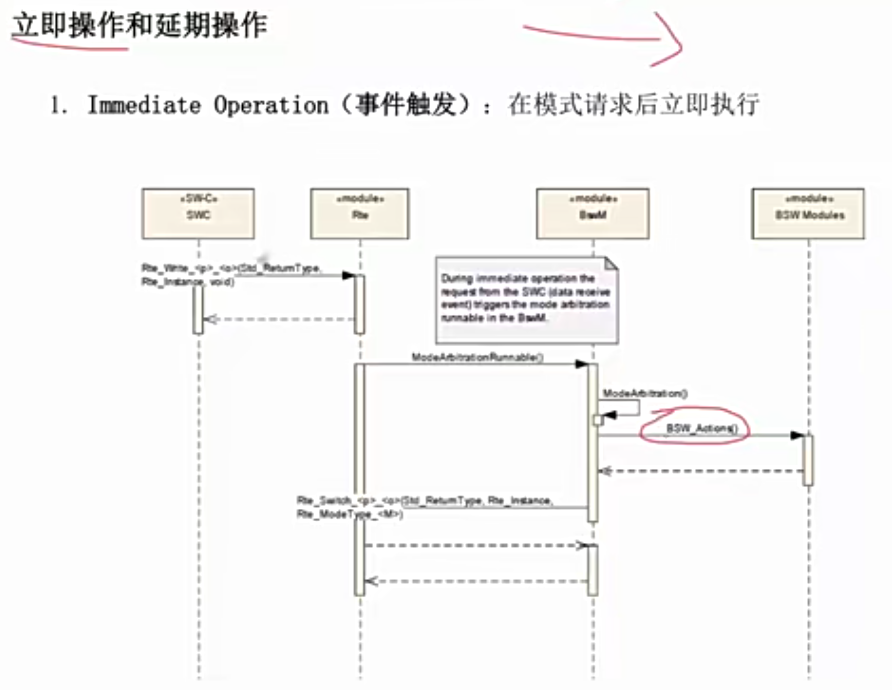
Mode Condition 2： B == 2

Logic Expression X = （Mode Condition1（逻辑运算符）Mode Condition 2）

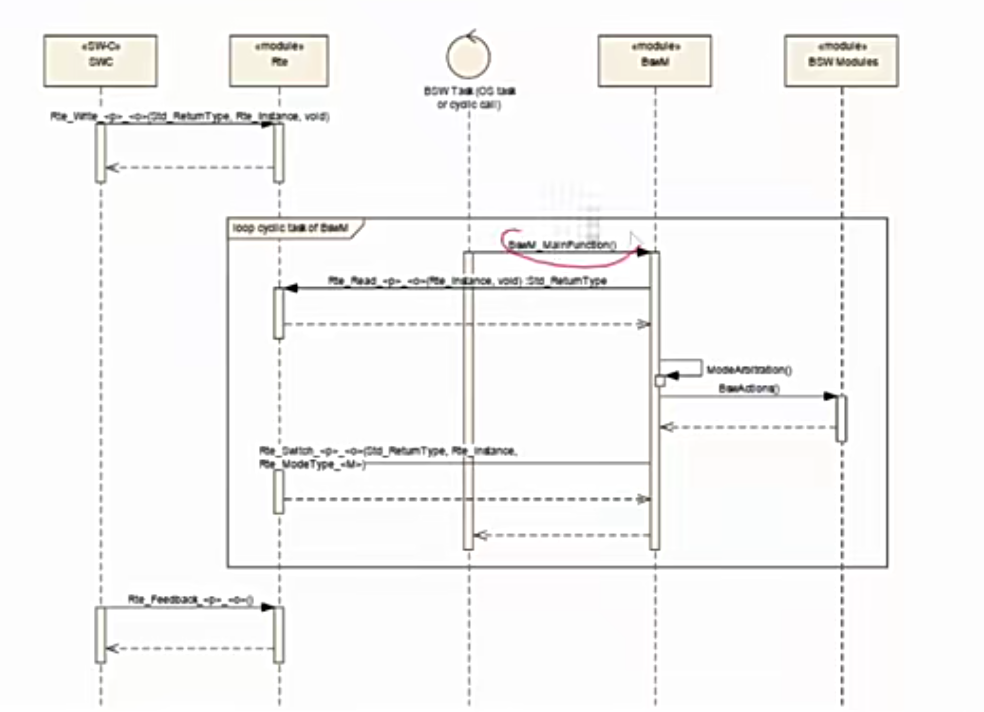
存在两种不同的方式来调度模式仲裁的处理：

**立即操作和延期操作**

1、**Immediate Operation（事件触发）**：在模式请求后立即执行

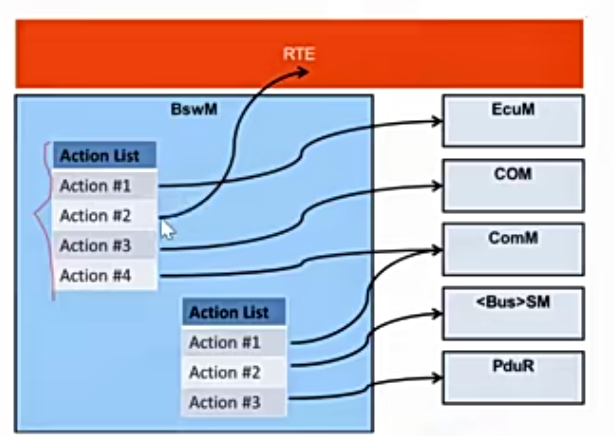


2.**Deferred Operation(轮询遍历)**：在模式请求后等到下一个BswM\_MainFunction中执行。



**模式控制**：

模式控制指的是基于模式仲裁的结果（TRUE or FALSE）执行相应的行为序列（Action）。这些序列称为“**Action List**”，且这些Action List会被BswM按顺序执行。



如上图所示，每一个Action List可以对应1个或者多个行为，Action List中的Action有下面三种类型：

直接调用其它BSW模块或者RTE模块的标准接口来实现一系列的控制（即**BSWM\_ATOMIC**）

ComM:设置对应通信接口的通信模式

COM：实现IPDU报文的切换

NM:开启或者关闭NM通信

链接其他Action List中的内容

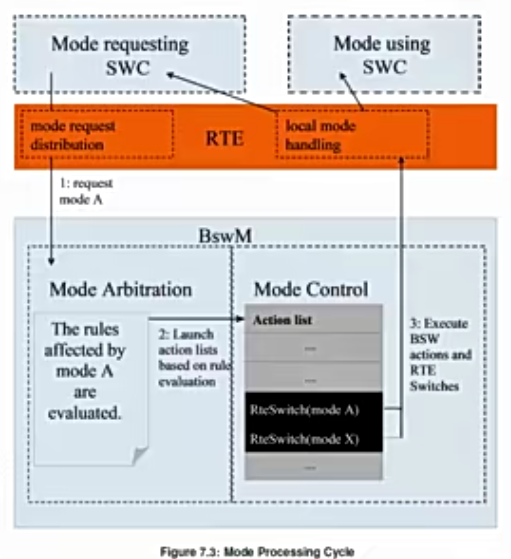
执行某些仲裁规则，当执行相应的操作列表时，将评估这些规则。

**模式控制流程**：

模式控制基本流程就是根据模式仲裁的结果去执行相应的Action List.

如下图所示，以SWC组件模式请求为例：

1. SWC通过RTE向BswM模块发送模式请求（Request Mode）
2. BswM根据已配置好的模式仲裁规则得出相应的仲裁结果
3. 仲裁结果关联的Action List
4. 其中Action List中时执行相应的Bsw相关的Action及RTE状态切换
5. 将模式处理通过RTE传到模式使用的SWC以及返回该模式到模式请求SWC



总结

BswM的核心逻辑代码就是：

BswM\_MainFunction()

{

if (expression == TRUE)

{

action\_true\_list();

}

else

{

action\_false\_list();

}

}

由于AutoSAR软件组件很多，**相互调用关系复杂**，如果没有BswM这样的模块统一管理，会使得**软件可读性差且容易产生Bug**，有了BswM的统一管理，让**代码一目了然，使得架构更加清晰**。