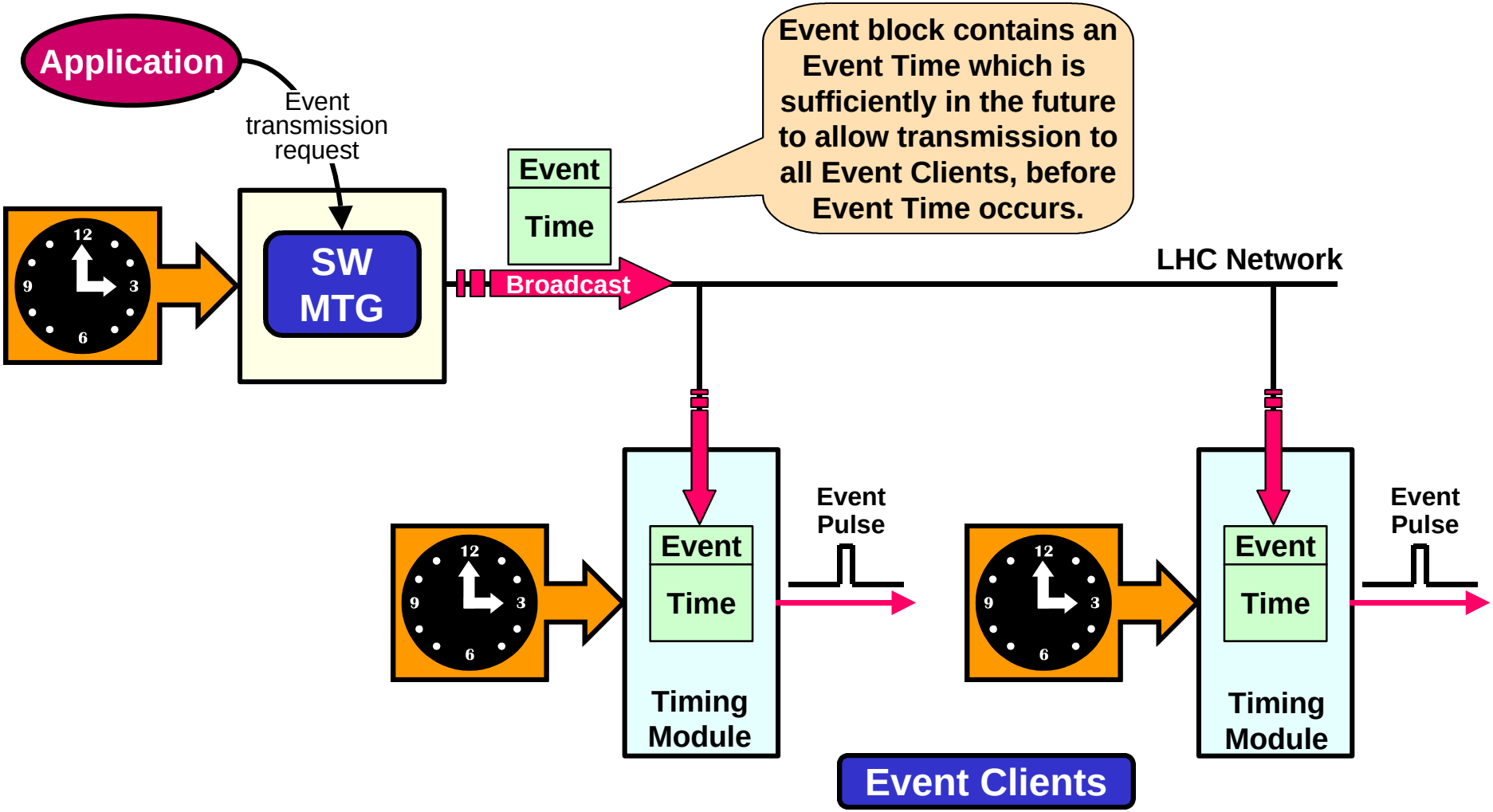
LHC Timing Events based on Absolute Time

Quentin King

Distribution of Absolute Time Events



LHC Timing Events

- **Synchronization for LHC will be important. There will be bad consequences if, for an event:**

    1. **some systems are triggered too early or too late, or**
    2. **some systems are not triggered at all.**

- **Thus for each system there will be a maximum tolerable jitter in the event response (1), which implies a maximum jitter in:**

    - The system's local Absolute Time source

    - The Event Time

    - The comparison between these two

- **In addition (2), if a system does not receive an event, or its local clock fails, the control system as a whole needs to know so that appropriate action can be taken.**

Absolute Time Distribution

- **Absolute time means uniquely defining “now”: Year, Month, Day, Hour, Minute, Second and of a Fraction of a second**
- **Requires compensation for geographical distribution**
- **Proposed primary source of Absolute Time: GPS/IRIG-B receivers (accuracy ~1μs)**
- **All LHC Systems will either directly connect to a GPS/IRIG-B receiver, or will be linked to one via a network or fieldbus.**
- **Systems will use their source of Absolute Time for event triggering *and* for time stamping logged data.**
- **For each system, the jitter in its source of Absolute Time must be low enough to meet its event and time stamping needs.**

LHC Event Block

Event Code

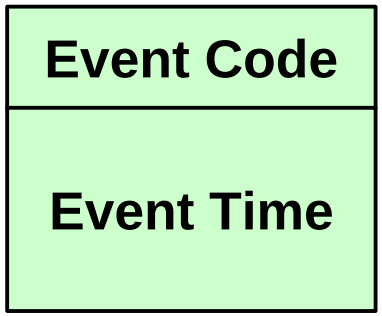
(4 bytes)

The Event Code could either be:

- **a number**
- **a bit mask**

A 4 byte number would provide an essentially limitless number of different events.

A 4 byte bit mask would provide only 32 different events, but a single Event Block could trigger any combination of these events simultaneously.



Event Time

(8 bytes)

The Event Time will use the Posix standard time structure:

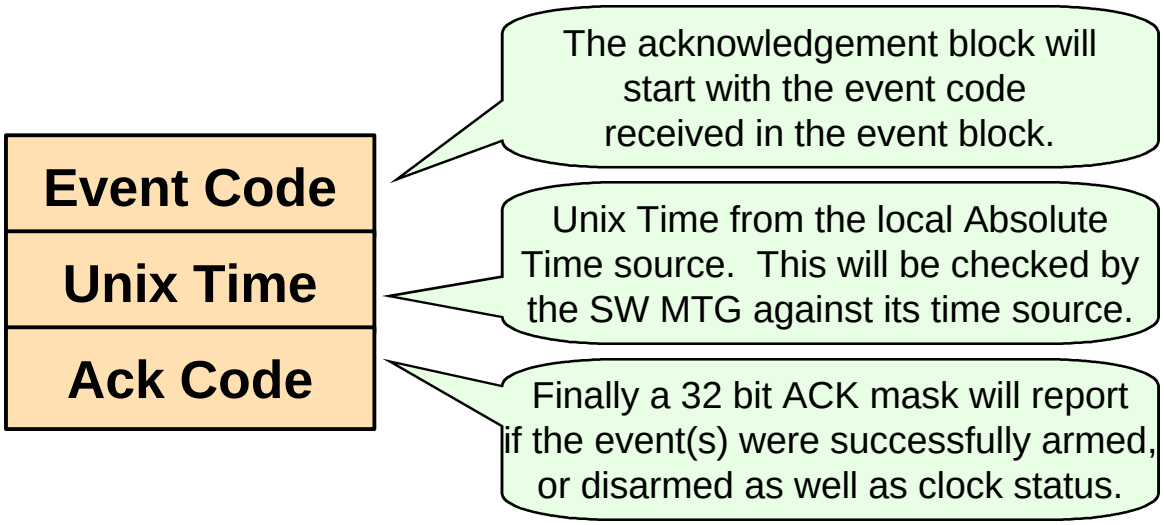
- **Unix time (4 bytes)**
- **ns within the second (4 bytes)**

Unix time gives the number of seconds since the start of 1970 and will work until 2106 (provided the value is unsigned).

An Event Time of zero would indicate that the event is to be cancelled.

Acknowledgement by Event Clients (1)

- **The SW MTG needs to know if all the Event Clients have received an Event Block, and if their local Absolute Time source are working.**
- **This requires communication from the clients back to the SW MTG.**
- **This communication will take the form of an Acknowledgement Block, sent in response to each Event Block received by the client. A possible format for the Acknowledgement Block is:**



Note: The identity of the Event Client does not need to be included in the Acknowledgement Block as it is included in the UDP message header.

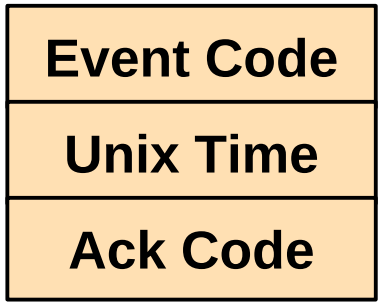
Acknowledgement by Event Clients (2)

- **When the SW MTG is asked to transmit an event, it helps to know in advance if:**

  - The Event Clients are alive and synchronized

  - The network to the Event Clients is working

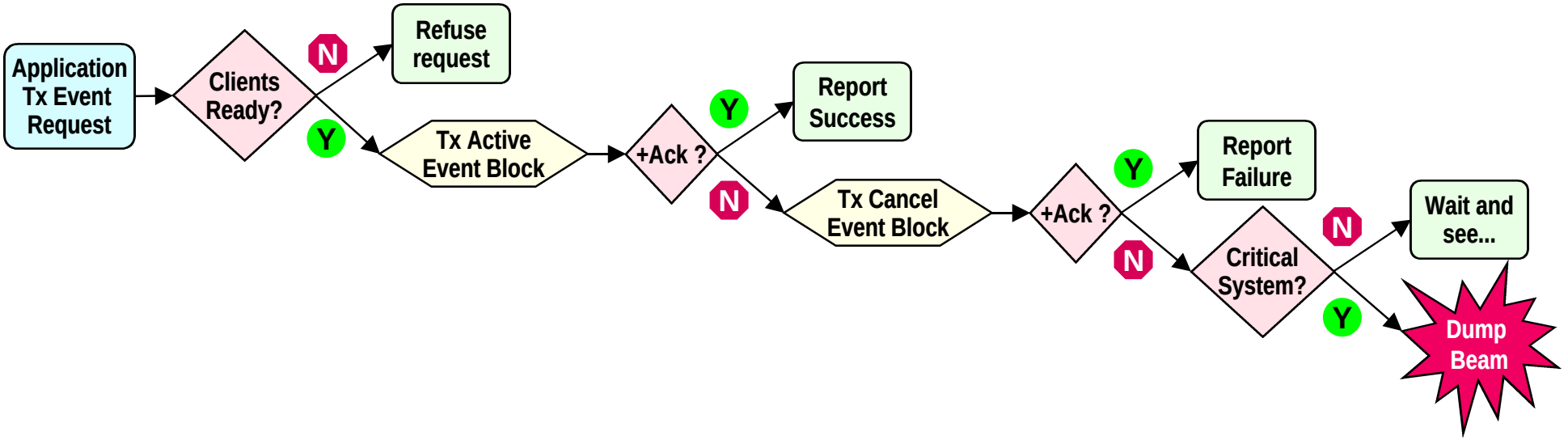
  This is possible if the SW MTG continuously sends NULL Event Blocks (zero Event Code) every few seconds.



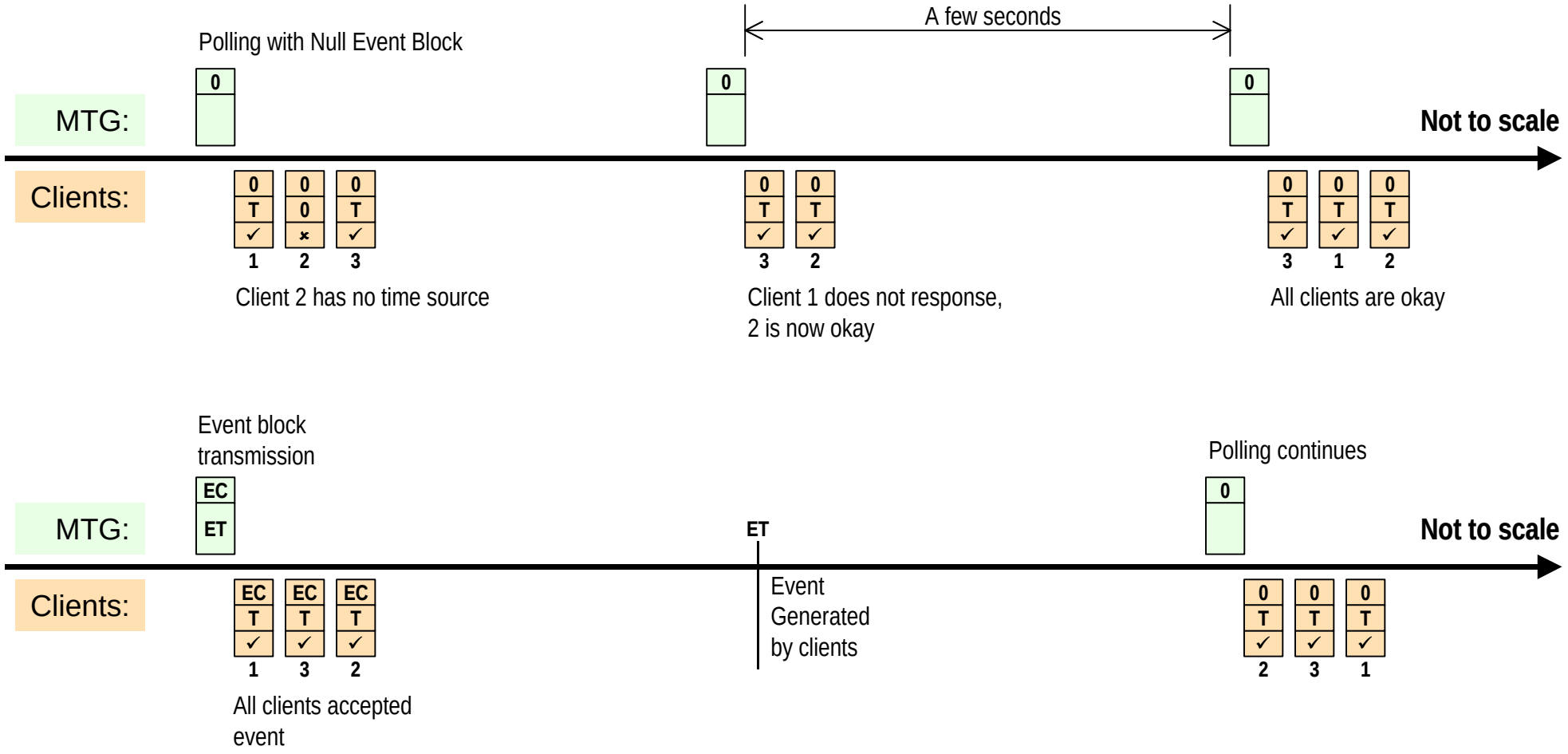
- **All the Event Clients will return an Event Acknowledgement Block in response to each NULL Event Block. In this way the SW MTG will know at all times which clients are alive and responding and have synchronized Absolute Time sources (to within a second).**
- **If the Event Code is a bit mask, then the Ack Code could be extended to be a 32 bit mask indicating which events are armed, followed by a second 32 bit status mask indicating the health of the local Absolute Time Source.**

SW MTG Decision Tree

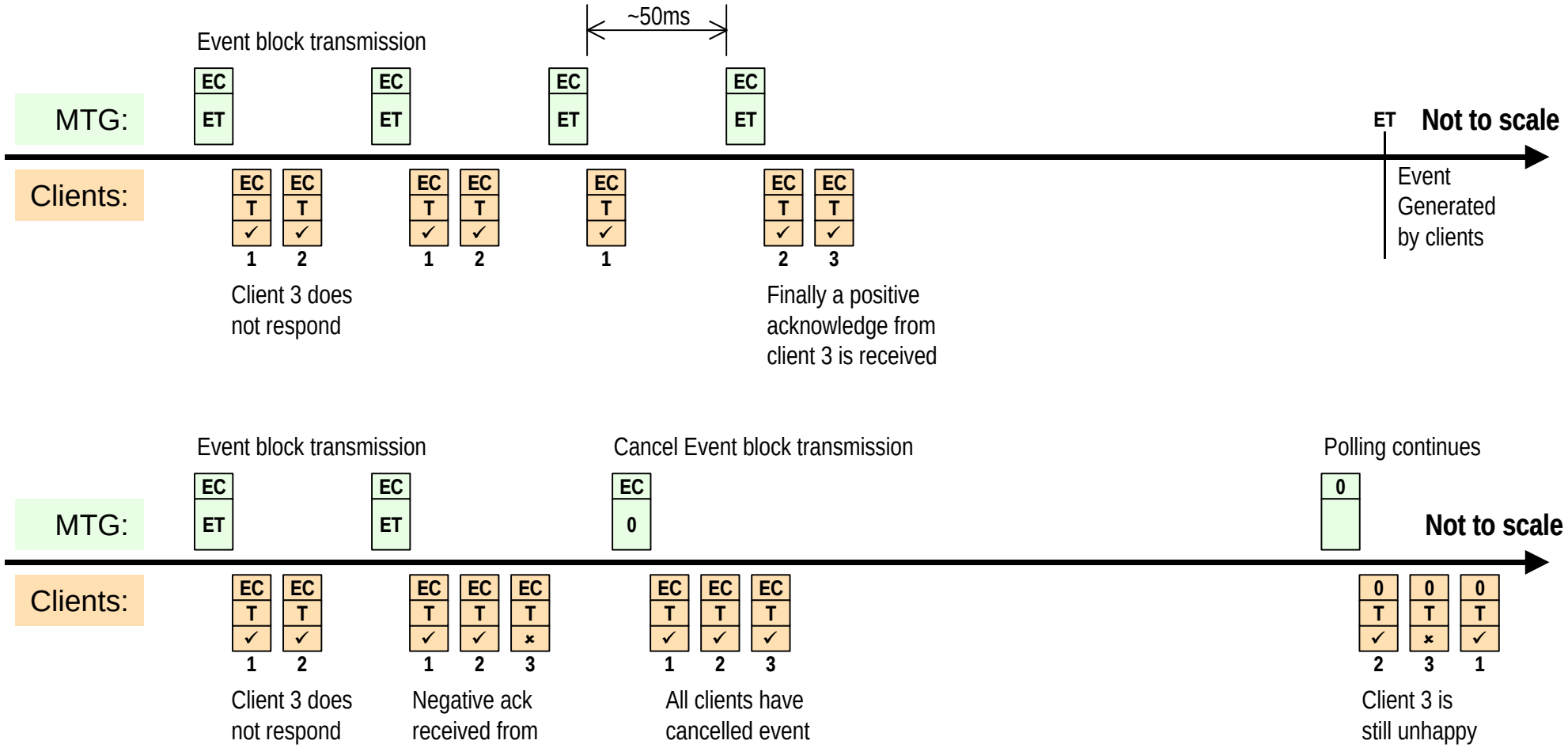
- **When an application requests transmission of an event (or events), the SW MTG will be responsible for dispatching the appropriate Event Block, checking that it is correctly received by all Event Clients, and in case of failure, dispatching a Cancel Event Block.**
- **The SW MTG will choose a time for the event sufficiently far in the future that this complete procedure can be accomplished before the event occurs.**



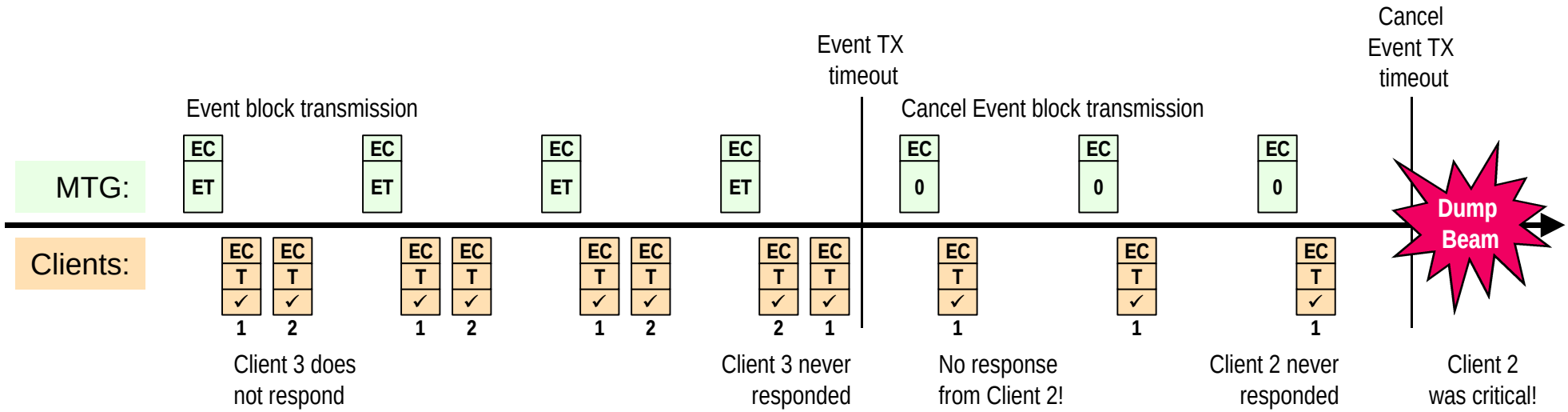
SW MTG ↔ Event Client Communications (1)



SW MTG ↔ Event Client Communications (2)



SW MTG ↔ Event Client Communications (3)



SW MTG Events and the Magnet Current Control System (MCCS)

- **The MCCS will have nearly 2000 systems, however, a top level Event Server will provide a single unified Event Client for the SW MTG.**
- **This MCCS Event Server will convert Event Blocks received from the SW MTG into MCCS broadcast commands using a predefined event command table.**
- **The success or failure of the broadcast “event command” will determine the contents of the Event Acknowledgement Block returned to the SW MTG.**

Communication Protocol

- **There will be several hundred Event Clients, so a broadcast transmission of Event Blocks is sensible using UDP and the same port number on each client system.**
- **Acknowledgement Blocks can also be returned using UDP with a single port on the SW MTG system receiving them all.**
- **UDP is not “secure” like TCP, but it is simpler and more efficient and by using a multiple transmission method (as shown earlier), reliable transmission of events and acknowledgements should be achieved.**
- **Alternatively, if native ATM is used instead of IP, then...**

Summary

- **If a suitable Absolute Time source is available for all Event Clients, then broadcasting a simple 12 byte Event Block in advance will allow synchronized operation of LHC systems.**
- **Distributing Absolute Time also allows uniform time stamping of logged data, which will be very valuable.**
- **A method of Event Client acknowledgement is necessary to make the system secure.**
- **The type of Event Code (number or bit mask) will need to be decided.**
- **A bit mask has advantages, provided a limited number of event types is sufficient.**