Concept of a Software Trigger for the TESLA Linear Collider experiment

Guenter Eckerlin
DESY, Hamburg

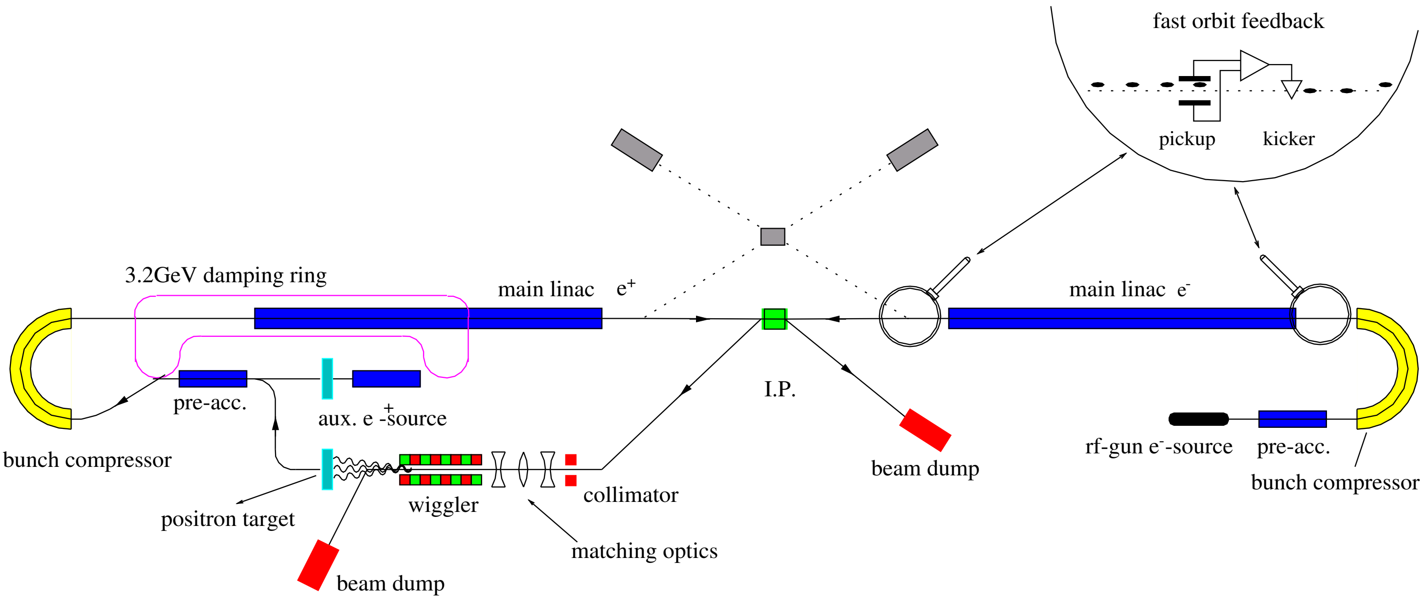
on behalf of

The ECFA/DESY Study Group

Outline

- Introduction
- Concept
- Feasibility
- Consequences
- Conclusions

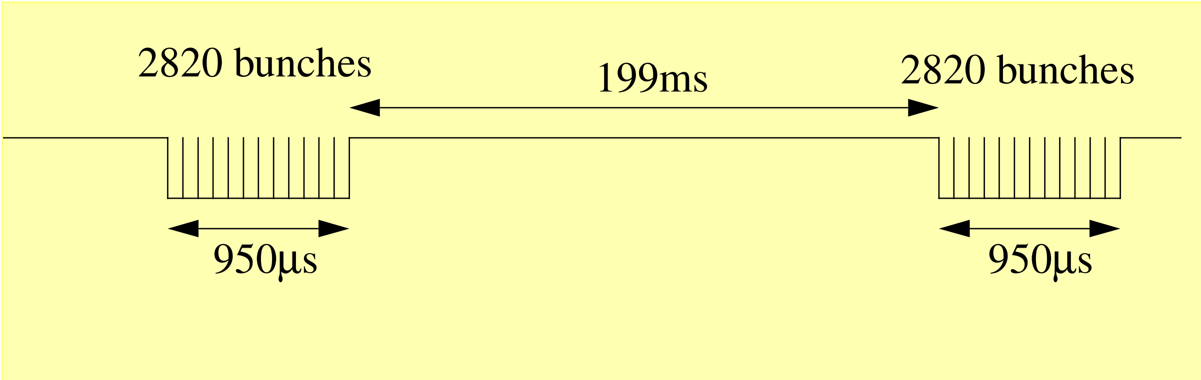
The TESLA Linear Collider



$e^+ e^-$ collider

Length	33 km
Centre of mass energy up to	800 GeV
Luminosity	$5x10^{34}$ $cm^{-2}s^{-1}$

The TESLA Operations Mode



parameters

trains with 2820 bunches at 5 Hz rate

bunch - bunch distance	337 ns
train length	950 µs
train - train distance	199 ms

All physics within 1 ms

-> deadtime free sampling

Nothing for 0,2 seconds

-> “long” time for online applications

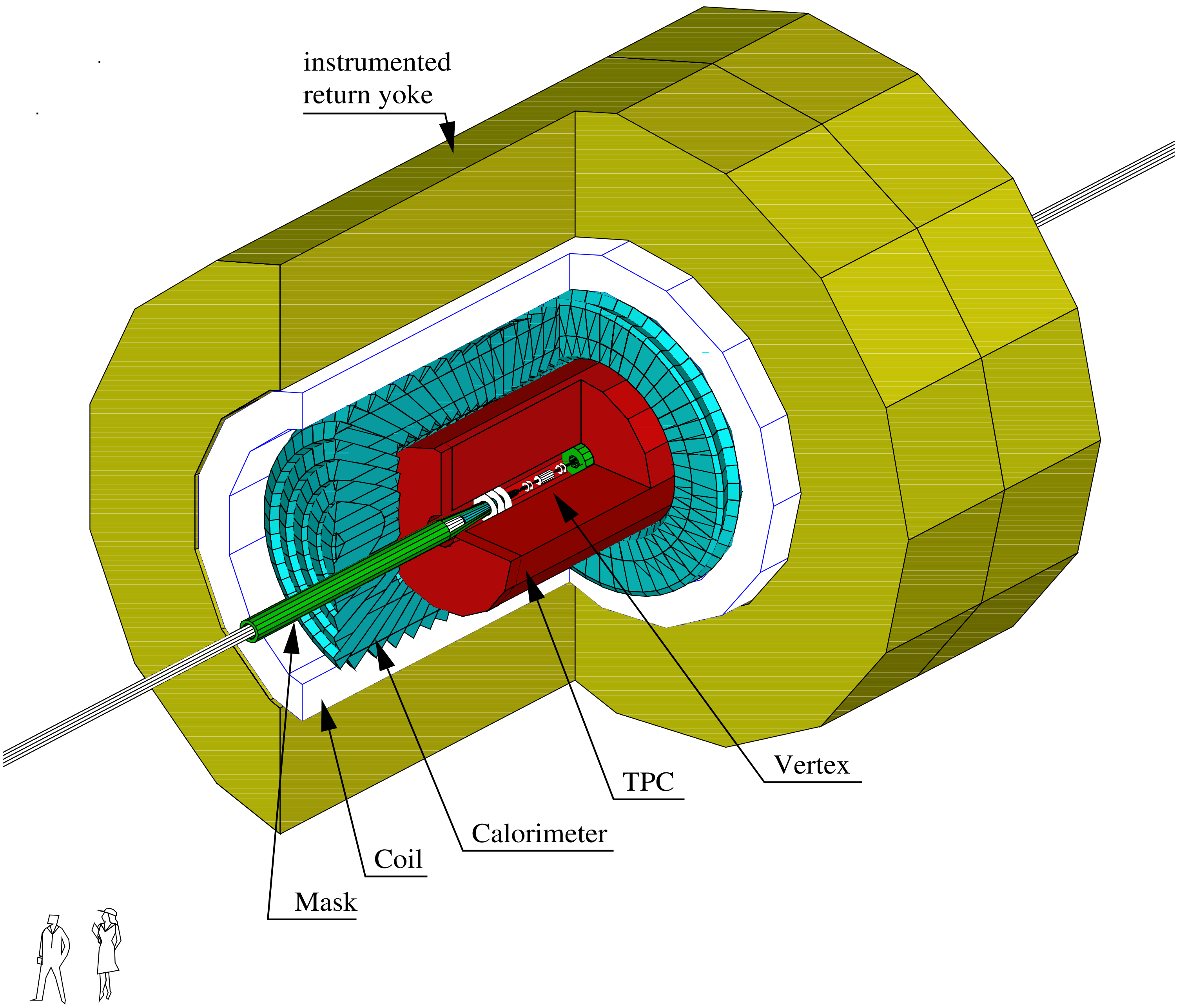
Concept of a Software Trigger

aim	requirement
• dead time free	sample complete train into pipeline without trigger interrupt. 1ms active front end pipeline.
• fully efficient and flexible	software event classification based on full event data available in train.
• manageable datavolume	
online	zero suppression / data compression.
offline	store classified events according to needs.
• easy maintenance ($\tau_{exp.}$ > 10 years)	off the shelf components. commonly used Operating System. high level programming language.
• scalability	modular system

Advantages

• commodity hardware	profit from industry standards and fast development in computing. large community helps debugging. large worldwide expertise. easy to extend and replace. cheaper than custom made.
• Operating system + high level language	good and debugged environment not dependent on specific hardware profit from future developments programming and debugging tools online = offline (debug once) same code = same person
• software trigger	full software control of the event selection, avoiding inefficiencies due to hardware triggers which act on data subsets.
• scalability	luminosity increase easy to adapt

A Detector for the TESLA Linear Collider



for details see talk from P.N. Burrows in parallel session 14 :
"A detector for precision study of high energy e^+e^- Annihil."

Physics and Background Rates
(at 800 GeV)

Physics events

• e^+e^--> e^+e^- + anything ("$\gamma\gamma$")	0.7/bx
• e^+e^--> anything	0.0002/bx

-> 2000 "$\gamma\gamma$" + 0.5 e^+e^- events per train

Background

• neutrons	1000n/bx
• e^+e^- pairs (Vertex Layer 1)	700hits/bx
• e^+e^- pairs (Vertex Layer 5)	60hits/bx
• photons from pairs (TPC)	1350γ/bx

-> background dominated

Data Volume

subdetector	channels $[10^3]$	hits/train $[10^6]$	MByte/train
Vertex	730.000	30	60
Forw. Tracker	20000	2	4
Interm. Tracker	10000	2	4
TPC	720	17	170
Calorimeter	200	8	32
Muon	200	1	4
LAT	10	1	4
		Total	**278**

Numbers for 800GeV and 3 Tesla solenoid field.

- 300 Mbyte/train
- physics event up to 1 MByte
- background events up to 100 KByte

-> looks feasible

Consequences

- 1ms active pipeline — digitising during the train to keep occupancy low.
  no active gating for the TPC.
- 200-300MB/0.2sec — high bandwidth (>> 2GByte/s) or multi train pipeline and parallel readout.
- processing in 200ms — parallel processing of trains.
  full data has to be available for whole train.

Conclusion

- A deadtime free readout for TESLA storing a full train (2820 BX) into a 1ms pipeline looks feasible.
- readout within 200ms using zero suppression or data compression is less demanding than for LHC.
- A software event filter on the basis of the full data in the train results in the highest possible efficiency for event selection. Rare physics and unfavourable event topology will be kept.
- Event data storage is completely under control of the event filter algorithms written according to the needs of the physics groups interested in the process.
- Commodity hardware, high level programming languages and a widely used OS will ease scalability and maintenance.
- Merging online and offline will result in a more effective use of resources, both person power and hardware.