Lattice Descriptions ("decks")

- are files containing element and beamline definitions which are used as input to lattice design and simulation programs such as MAD

- contain the usual definitions of:
  - magnets
  - rf devices (damping ring cavities, L-band, S-band, and X-band structures)
  - drifts

- include explicit definitions of:
  - diagnostics (bpms, wire scanners, current monitors, bunch length monitors, synchrotron light ports, etc.)
  - correction devices (corrector dipoles, trim windings, feedback actuators, magnet movers, girder movers)
  - collimators, abort systems, dumps

- contain *treaty point* definitions of:
  - Twiss parameters
  - beam parameters
  - global coordinate system position and orientation
WBS Level 3 Areas

For a description of the NLC Global Coordinate System, see:
http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Lattices/Documentation/names.html

For descriptions of the lattice elements contained in each deck, see:
http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Lattices/Documentation/Subsystem%20Descriptions.doc
Working toward CDR: Strategy

- convert decks to a standardized input format
- modify selected optics programs to read standardized input files and generate standardized output files
- automate the process of transferring lattice data directly into the engineering and costing database
- establish a formal protocol for controlling the release/revision of decks and related files
- fine tune the optics design with defined milestones for critical design decisions
Status of NLC Decks

A complete set of decks defining the CD-1 version of the NLC lattice exists. As part of the CDR process we are presently completing work on the next version of the lattice which, aside from continuing work on the collimation systems and post-IP dump lines, will be close to the final version.

General modifications to the decks include:

- convert decks to *Extended Standard Input Format* (XSIF) where necessary

- substitute standardized magnet types wherever possible to reduce cost

- add beam instrumentation, diagnostic and correction devices, collimators, abort systems, *etc.* where needed

- assign NLC-wide names to beamline devices
Status of NLC Decks (cont.)

The CD-1 lattice already contains some major changes to the ZDR lattice, including:

- **main linacs**
  - 3 X-band structures per girder, consistent with DLDS
  - reworked diagnostics sections

- **beam delivery systems**
  - new final transformer and final doublet
  - new dump line lattices
  - staggered layout for two IPs

- **injector systems**
  - new damping ring designs (main and pre-DR)
  - new lattice designs for low frequency linacs (drive, boosters, and pre-linacs)

The remaining work to be done is mainly in the injector systems, consisting of rematching the existing transport lines to the redesigned beamline sections and completing the general modifications mentioned earlier.
Deck Management

• optics designers
  – Tor Raubenheimer (everything)
  – Peter Tenenbaum (beam delivery systems)
  – Yuri Nosochkov (main linacs, dump lines)
  – Zenghai Li (boosters, pre-linacs)
  – Paul Emma (damping rings, bunch compressors, beam delivery systems)
  – Dick Helm (collimation, beam delivery systems)
  – others ...

• “Deck Masters”
  – Peter Tenenbaum (beam delivery systems)
  – Mark Woodley (injection systems, main linacs)

A document, *Beamline Decks Coding Standards*\(^1\), which defines what goes into NLC decks, exists.

\(^1\) See the unix file: /afs/slac/g/nlc/lattice/documentation/deckstandards.ps
Deck Management (cont.)

- decks and related files will be managed by the Deck Masters using the Concurrent Version System\(^1\) software

- full lattice *releases* will take place every six months; lattice *revisions* to occur between releases as needed

- a committee, comprised of Deck Masters, area managers, and selected TSET-team members, will sign off on each revision/release

- a history file which describes the changes will be maintained for each deck

- read-only access to all releases/revisions will be available to both local and remote users via WWW

---

\(^1\) See *Version Management with CVS (version 1.10)*, by P. Cederqvist, *et al.*, and/or the Cyclic Software WWW page at [http://www.cyclic.com](http://www.cyclic.com) for details.
Extended Standard Input Format (XSIF)

- start with *Standard Input Format*\(^1\), which includes specifications for the definition of beamlines and of element types:
  - DRIFT, SBEND, QUADRUPOLE, SEXTUPOLE, OCTUPOLE, MULTIPOLE, SOLENOID, RFCAVITY, ECOLLIMATOR, RCOLLIMATOR, HKICKER, VKICKER, HMONITOR, VMONITOR, MONITOR, MARKER, SROT

- define some additional element types:
  - LCAVITY, INSTRUMENT, PROFILE, WIRE, IMONITOR, BLMONITOR, SLMONITOR, YROT

- add **Aperture** attribute to most elements

---

PT has created **libxsif**\(^2\), a standalone library for parsing NLC XSIF beamline decks. This library was successfully used to make the program LIAR capable of reading XSIF files.

---


\(^2\) See: [http://www-project.slac.stanford.edu/lc/local/AccelPhysics/codes/xsif/README.txt](http://www-project.slac.stanford.edu/lc/local/AccelPhysics/codes/xsif/README.txt)
Extended Tape File Format (XTFF)

• a set of standardized output file formats, derived from MAD tape file definitions\(^1\) and modified to suit the requirements of NLC

• start with MAD tape file format definitions, which include specifications for element attribute data to be included in all tape files, as well as specific data to be included in each type of tape file (\texttt{SURVEY}, \texttt{TWISS})

• add aperture, beam energy, \textit{engineering name}, and Formal Device Name to element attribute data blocks

• define a new tape file type, \texttt{ENVELOPE}, which outputs beam sizes to allow checking of beam-stay-clear

NLC Device Names

• each beamline device in the decks will be assigned a unique Formal Device Name (FDN)

• FDNs consists of three fields:
  – area
  – function
  – location

• the *area field* gives the name of the area to which the device belongs; for beamline components, this is the name of the deck in which the device appears

• the *function field* is descriptive of what the device is or what it does; for beamline components, this will be something like QUAD, BEND, or BPM

• the *location field* is a number related to the location of the device; for beamline components, this will be the distance along the beamline from the start of the deck in which the device appears

• in addition, most devices will also have an *engineering name* which appears in the standard output files and is used for costing in the WBS
XTFF/CAD Interface

Layout drawing for the Main Damping Ring created by Solid Edge CAD program from XTFF file
NLC Database WWW Interface

Welcome to the NLC Database

NLC Device Database Queries
- Formal Device Name Query*
- Area Name Query*
- Device Type Query*
- Engineering Type Query*

NLC Device Database Maintenance
- Area, Device & Engineering Name Maintenance Page

NLC Database Documents
- Formal Device Name Format by Woodley, Rabinheimer and Sams on October 27, 1998

*Links followed by an asterisk are limited to the SLAC community.
NLC Database WWW Interface

Results of Formal Device Name Query downloaded directly into Excel spreadsheet
Simulation Codes

Many programs are being used for NLC lattice design and simulation. Where necessary, these programs have been adapted to run on multiple computing platforms including Windows NT, unix (AIX and Solaris), and VMS. In addition, some of these programs have had functional modifications to make them more suitable for NLC use.

• MAD (lattice design, tracking)
  – started with version 8.23 from CERN
  – reads XSIF; writes XTFF
  – added acceleration (LCAVITY element type)
  – new ENVELOPE tape file type
  – coherent synchrotron radiation effects in tracking

• DIMAD (lattice design, tracking)
  – reads XSIF; writes XTFF
  – new PHASE command for phasing linacs
  – “twin-horned” energy distribution for tracking
  – extended chromatic precision option for tracking
  – fitting of tracking results

• LIAR (tracking in linacs with wakefields)
  – reads XSIF
  – callable as a function from within Matlab
  – used for linac feedback simulations

Simulation Codes (cont.)

- **TRANSPORT** (lattice design)
  - 3rd order version from FERMILAB

- **TURTLE** (tracking)
  - 3rd order version from FERMILAB
  - locally modified for SLC use by Tracy Usher

- **GUINEAPIG** (e+e- beam-beam simulation)
  - includes disruption, beamstrahlung, pair production
  - used for simulating luminosity, detector backgrounds

- **EGS** (electron gamma shower simulation)
  - used for e+ production system design
  - see home page: [http://ehsun.lbl.gov/egs/egs.html](http://ehsun.lbl.gov/egs/egs.html)

- **PARMELA** (gun/injector simulation)
  - used for e- source design

- electromagnetic modeling codes developed and maintained by the Numerical Modeling Group

- others ...
Simulation Codes: Management Issues

• version control
  – very informal at present
  – plan to use CVS for locally maintained optics codes

• benchmarking/verification
  – very informal at present
  – need formal protocol
  – need test decks with outputs

• distribution
  – local use (unix/NT) from remotely mounted shared NLC group disks
  – offsite use via anonymous ftp
Summary

• the infrastructure for managing decks and lattice releases is in place, including standardization of input and output files, NLC-wide device naming, and lots of online documentation

• a suite of lattice design and simulation codes is in place, modified for NLC where needed, and available on multiple computing platforms

• some tools for automatically transferring lattice data to the engineering and costing groups exist, with more on the way

• the post CD-1 lattice release is mostly complete

• what to do next is clearly defined and presents no major challenges
Lattice Descriptions, Simulation Codes, and Related Documentation for NLC

Appendix
NLC e- Injection System Decks (ZDR)

```
“Treaty points”: $E, \gamma, \beta, \alpha, \eta, \eta', \sigma_z, \sigma_E, X, Y, Z, \theta, \phi, \psi$

- ESRC: transport to booster linac (80 MeV)
- EBSTR: S-band booster linac (to 1.98 GeV)
- ELTR: spin rotator and transport to damping ring
- EDR: main damping ring
- EBC1: ring extraction, spin rotator, and first bunch compressor (“wiggler”)
- EPLIN: S-band pre-linac (to 10 GeV)
- EBC2: second bunch compressor (180° arc, 1 GeV X-band rf, chicane)

→ tune-up dump lines not shown (in ELTR, end of EBC1, end of EPLIN, end of EBC2)
```
• PESRC : drive e- transport to booster linac (80 MeV)
• PELIN : drive e- S-band booster linac (to 6.22 GeV)
• PETLn : drive e- transport lines to e+ production targets
• PCAPn : e+ capture sections and transport lines to booster linac (250 MeV)
• PBSTR : S-band booster linac (to 1.98 GeV)
• PLTR : transport to pre-damping ring
• PPDR : pre-damping ring
• PXFER : pre-damping ring extraction and transport to main damping ring
• PDR : main damping ring
• PBC1 : ring extraction, spin rotator, and first bunch compressor (“wiggler”)
• PPLIN : S-band pre-linac (to 10 GeV)
• PBC2 : second bunch compressor (180° arc, 1 GeV X-band rf, chicane)

→ tune-up dump lines not shown (end of PELIN, in PLTR, end of PBC1, end of PPLIN, end of PBC2)
NLC Beam Delivery System Decks (CD1)

- **ECOL**: e- post-linac collimation (500 GeV)
- **EIRTn**: e- transport and final focus systems
- **EDLn**: post-IP e- dump line
- **PCOL**: e+ post-linac collimation (500 GeV)
- **PIRTn**: e+ transport and final focus systems
- **PDLn**: post-IP e+ dump line

→ tune-up/abort dump lines not shown (end of main linacs (2), start of final focus systems (4))
1 Introduction

The top-level description of every beamline in the Next Linear Collider is, or will be, a "master option deck" in some sense, a kind of database of the Standard Input Format [5]. This master deck will ultimately be used for beam-line simulations, mechanical layout of the XLC, generation of the master hardware database, accelerator control system functions, and other functions we cannot as yet foresee.

In order to simplify the life of the various end-users, it is helpful to keep certain features of the decks consistent from region to region. These Standards are an attempt to distill the features which are of greatest importance to end-users other than the Deck-Masters of each region. In addition, this Standard is a guide to resolving conflicts between the versions of Standard Input Format in use today, and sets minimum requirements on the input parsers of all "NLC versions" of programs using the Standard Input Format.

2 Language

The language which describes each beamline shall be the Standard Input Format, as it is variously (and somewhat controversially) described in [1, 2, 3], plus some additions and extensions (apertures, linear accelerator structures, etc); the language with NLC extensions is known as the Extended Standard Input Format (ESIF).

The master deck may contain only beamline and element description information, no action commands are permitted except "USIF" and "!FINISH".

2.1 Units

The unit keywords used by USIF (i.e., USIF/STANDARD, USIF/UMAD, USIF/UTRANSPORT) are forbidden; consequently all decks must use the standard units set. See Appendix A for enumeration of the Standard units set.

2.2 Elements and Element Names

Each element's deck name (hereforth called its option name) shall be from 1 to 8 characters in length, where character 1 must be a letter, and characters 2 through 8 may be:

- a letter
- a digit
- a period (".")
- an underscore ("_")
- or a dollar sign ("\$")

Names must not be-case sensitive (i.e., a given beamline may not contain two different devices whose names are the same, or different, case).

Elements may be re-used an arbitrary number of times in a deck.

2.3 Engineering Type Specification

The engineering specification of a device's properties (hereforth called its engineering type) is a string of up to 31 characters specified by the TYPE keyword. The first character must be either a letter or a digit; subsequent characters may be letters, digits, periods, underscores, dollar signs, or colons (":"). Consecutive engineering names are prohibited.

See: /afs/slac/g/nlc/lattice/documentation/deckstandards.ps
WWW Documentation

JLC / NLC Accelerator Physics at SLAC

Definition of Extended Tape File Format (XTFF)

Output files from various optics programs will be used as input to other programs that do, for example, automated CAD element layout, parts counting, and database generation. In order to minimize the number of special programs needed to convert the output of optics programs into a form suitable for these other tasks, we define a standard format for some types of output generated by the optics programs we are using for NLC.

We have chosen the MAD "tape file format", used for SURVEY and TWISS output, as the basis of our standard, and have added some extensions to create what we call extended tape file format (XTFF).

MAD output SURVEY and TWISS data to "tape files" as follows:

- the "tape files" are ASCII (readable by humans)
- the data for each element is output as a block of 50 column lines
- the first three lines of each block contain element definition data, and the remaining lines (2 for SURVEY and 3 for TWISS) contain coordinate or tune parameter data for that element

XTFF does not change the format of columns 1-80, but appends additional data to the first two lines (referred to in the MAD manual as the Common Output position records) in columns 81-132, as follows:

- APERTURE, TYPIC (the element's engineering name), and beam energy are appended to line 1
- the element's Formal Device Name (FDN) is appended to line 2

Table 1 lists the XTFF contents of the first position record for the element keywords allowed in XSE for NLC. The columns locations of each data field is noted at the top. The FORTRAN and C-language format specifications which can be used to read the first position record are:

- FORTRAN format: (A9, A30, F12.6, 4F10.9, 1X, A10, 1X, E10.9)
- C-language format: ("")

Table 2 lists the XTFF contents of the second position record. The FORTRAN and C-language formats which can be used to read the second position record are:

- FORTRAN format: (5F16.9, 2X, A24)
- C-language format: ("")

Next Linear Collider

WWW Documentation

JLC / NLC Accelerator Physics at SLAC

NLC Device Naming Convention

M. Woodley, T. Raubenheimer, R. Sass

October 27, 1998

The purpose of this note is to describe the formal global naming scheme for NLC devices. To begin with, the proposed scheme is intended for use in naming NLC beam line components (magnets, accelerators, diagnostics, etc.), however the scheme should be flexible enough to be extended to other NLC components.

The global naming scheme makes references to individual NLC devices unambiguous, and allows us to enumerate global device lists for engineering and cost estimates. The unique name of each device will provide a first level key into the databases that will be used for cost estimation and scheduling, as well as providing the essential tag needed during production and installation, and by the future NLC control system. Because the name includes information on what a device is and where it is located, lists sorted by device type or geographical region can be constructed by keying off portions of the name.

A global device name consists of three fields as follows:

- **area field**: a beam line area designation (e.g. EDR or FF1)
- **function field**: a generic functional device descriptor (e.g. QUAD or HCOR)
- **location field**: a number related to a device's location along the beam line.

To improve the readability of the names a colon (:) is used to separate the fields. At this time there is no limit on the number of character positions assigned to each field, although brevity is desirable. For consistency, all device names are upper case, although we should avoid making the names case sensitive. Each field consists of alphanumeric characters (A-Z, 0-9); the special characters "-" (period) and "_" (underscore) will also be allowed. An example of a name:

EDR:QUAD:234

Ideally, the location field is related to the longitudinal position of the device along the beam line.

See: http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Lattices/Documentation/NLC_Component_Catalog.ps
WWW Documentation

JLC / NLC Accelerator Physics at SLAC

NLC Lattices

- Documentation
- ZDR ReadMe (deck status, locations, and lengths)
- ZDR transport decks and output
- ZDR optics plots (pdf, ps, and topdraw)
- CD1 ReadMe
- CD1 decks and output
- CD1 optics plots (pdf, ps, and topdraw)
- CD1 costing recommendations

See: http://www-project.slac.stanford.edu/lc/local/AccelPhysics/nlc_lattices_index.htm
JLC / NLC Accelerator Physics at SLAC

NLC Lattice Documentation

- NLC Lattice Plans
- NLC Lattice Issues
- CD1 Beamline Descriptions
- NLC Device Naming Convention
- A Global Coordinate System for NLC
- NLC Beamline Decks Coding Standards
- Definition of Extended Tape File Format (XTFF)
- NLC Component Catalog

See: http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Lattices/Documentation/documentation_index.htm
NLC Global Coordinate System

$X, Y, Z, \theta$ (yaw), $\phi$ (pitch), $\psi$ (roll)

- Positrons: $X > 0$, $Z > 0$
- Electrons: $X > 0$, $Z < 0$

$\theta > 0$

$\phi > 0$

$\psi > 0$
JLC / NLC Accelerator Physics at SLAC

A Global Coordinate System for NLC

The location and orientation of NLC devices will be defined by 6 coordinates (X, Y, Z, θ, φ, ψ) expressed in a global right-handed Cartesian coordinate system. Note that θ is "yaw", φ is "pitch", and ψ is "roll". The units for the position coordinates X, Y, and Z will be meters, while the units of the orientation angles θ, φ, and ψ will be radians.

The NLC main linacs will lie on the Z-axis, with the polarized electron source located at negative Z and the positron source at positive Z. The origin of the X-, Y-, and Z-axes will lie half way between the NLC IP's.

To illustrate the relationship between the NLC accelerator complex and this global coordinate system, imagine that NLC has been constructed with a north-south orientation (i.e. the main linacs lie on a north-south meridian). The Z-axis of the global coordinate system will then be oriented north-south, with positive Z toward the north. The X-axis would be oriented east-west, with positive X toward the west. The Y-axis would be oriented up-down, parallel to gravity at the origin, with positive Y up. The main linacs will lie along the Z-axis, with the electron injector system located at negative Z (the southern end) and the positron injector system located at positive Z (the northern end). The origin of the coordinate system would defined by the intersection of the Z-axis and a line connecting the two IP's. For the staggered IP geometry that has been adopted, the two IP's would be equidistant from the X-axis, IP #1 would be located at negative Z and negative X, with IP #2 at positive Z and positive X. Both the electron and positron injector systems would be located at positive X (west of the main linacs).

The positive Z-axis defines the θ = 0 direction, with θ > 0 toward positive X, θ < 0 toward negative X, and -π < θ ≤ π. The X-Z plane defines the φ = 0 direction, with φ > 0 toward positive Y, φ < 0 toward negative Y, and -π < φ ≤ π. Finally, the roll angle ψ for a device is positive when that device, as seen by the beam, is oriented clockwise about its longitudinal axis, with -π < ψ ≤ π.

Note that new releases of the NLC lattice design will (probably) entail changes to the absolute coordinates of NLC devices; we anticipate such changes to occur at approximately 6 month intervals.

Created: 08/14/98 by Mark Woodley
Last modified: 10/25/98 by Mark Woodley

See: http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Lattices/Documentation/NLC_Global_Coordinate_System.htm
WWW Documentation

JLC / NLC Accelerator Physics at SLAC

NLC Simulation Codes

The UNIX versions of these programs can be found at http://www-project.slac.stanford.edu/lc/local/AccelPhysics/Codes while the NT versions of these codes can be found in the AccelPhysics/Codes folder in the NLC www-project area.

MAD

The NLC version of MAD (v8.23/acc) is available for UNIX/AIX workstations (maintained by Tor Raybender) and for Windows NT (maintained by Mark Woodley). This version is XSSP/XTFF compliant and includes linear acceleration. See the MAD Readme document for details.

DIMAD

The NLC version of DMinD (v2.04 Fortran-90) is available for AIX workstations and for Windows NT (maintained by Peter Tenerbaum). See PTA's DIMAD Readme, DIMAD Charges, and DIMAD Features documents, and the Dimad User's Guide, for more details.

LIAR

Fortran simulation of 6-D phase space, written by Ralph Assmann and others. LIAR v2.2 is available for AIX and S-Linux workstations, Windows NT, and VMS (all maintained by Peter Tenerbaum). See PTA's LIAR Readme and LIAR Charges documents for more details. A VMS Matlab-callable version of LIAR is also available (maintained by Linda Hendrickson).

TRANSPORT

3rd order TRANSPORT (v19941020) is available for AIX workstations direct from Fermilab. A Windows NT version is also available (maintained by Jim Turner).

TURTLE

3rd order TURTLE (v19930121 with local enhancements), is available for Windows NT and VMS systems (maintained by Tracy Usher).