\begin{verbatim}
result in(stateT& state,
    const externT* from, const externT* from_end, const externT*& from_next,
    internT*      to,    internT* to_limit,    internT*& to_next) const;

and in 22.2.1.5 and 22.2.1.5.2 [lib.locale.codecvt.virtuals], replace
member do_convert with corresponding members do_in and do_out, the same
way. do_out() is defined identically as the old do_convert(); do_in
is defined identically with the exception of the substitution of "extern"
for "from" and "intern" for "to" in the description; nonvirtual members
in() and out() simply forward to do_in() and do_out(). In 22.2.1.6
[lib.locale.codecvt.byname], replace member do_convert() with do_in()
and do_out().

In Clause 22, Table 2 (Locale Category Facets), remove the entry for
    codecvt<char,wchar_t,mbstate_t>

In 27.8.1.1 [lib.filebuf], in paragraph 4, replace the specification
of how conversions are performed to say:

Specifically, the facet to use is obtained 'as if' by

    const codecvt<charT,char,typename traits::state_type>&
    a_codecvt =
    use_facet<codecvt<charT,char,typename traits::state_type>>(getloc());

The formal name a_codecvt is used in descriptions of virtuals underflow
and overflow.

In 27.8.1.4 [lib.filebuf.virtuals], in the description of member
underflow, change the definitive code example as follows:

    char extern_buf[XSIZE];
    char* extern_end;
    internT intern_buf[ISIZE];
    internT* intern_end;

    codecvt_base::result r =
    a_codecvt.in(st,extern_buf, extern_buf+XSIZE, extern_end,
                   intern_buf, intern_buf+ISIZE, intern_end);

In the description of member overflow, change the definitive code
example to use the call as follows:

    internT* int_end;
    char xbuf[XSIZE];
    char* xbuf_end;

    codecvt_base::result r =
    a_codecvt.out(st, pbase(), pptr(), int_end, xbuf, xbuf+XSIZE, xbuf_end);

--------

In 22.2.1.5   [lib.locale.codecvt],
    22.2.1.5.1 [lib.locale.codecvt.members],
    22.2.1.5.2 [lib.locale.codecvt.virtuals], and also where appropriate in
    22.2.1.6   [lib.locale.codecvt.byname],

add the following member functions to the locale codecvt<> and
codecvt_byname<> facets.

Add the nonvirtual public member:
\end{verbatim}
int encoding() const throw();

Returns: do_encoding(); and add the virtual protected member:

int do_encoding() const ...

constant number of external character(s) needed to produce an internal character; or 0 if this number is not a constant.